Apple IIc User Guide

Gary Phillips
Donald Scellato

APPLE //c

APPLE //c

User Guide

Gary Phillips
Donald Scellato

Brady Communications Company, Inc.
A Prentice-Hall Publishing Company
Bowie, MD

Apple //c User Guide

Library of Congress Cataloging in Publication Data

Phillips, Gary.
Apple IIc user guide.

Includes index.
1. Apple IIc (Computer) I. Scellato, Donald J. II. Title.
QA76.8.A662P48 1984 001.64 84-11156

ISBN 0-89303-307-3

Prentice-Hall International, Inc., London
Prentice-Hall Canada, Inc., Scarborough, Ontario
Prentice-Hall of Australia, Pty., Ltd., Sydney
Prentice-Hall of India Private Limited, New Delhi
Prentice-Hall of Japan, Inc., Tokyo
Prentice-Hall of Southeast Asia Pte. Ltd., Singapore
Whitehall Books, Limited, Petone, New Zealand
Editora Prentice-Hall Do Brasil LTDA., Rio de Janeiro

Printed in the United States of America

84 85 86 87 88 89 90 91 92 93 94 1 2 3 4 5 6 7 8 9 10

Publishing Director: David Culverwell
Acquisitions Editor: Chris Williams
Production Editor/Text Design: Michael J. Rogers
Art Director/Cover Design: Don Sellers
Assistant Art Director: Bernard Vervin
Manufacturing Director: John Komsa

Typesetting: Alexander Typesetting, Inc., Indianapolis, IN
Printing: Fairfield Graphics, Fairfield, PA
Indexing: William O. Lively

CONTENTS

Note to Authors

Do you have a manuscript or software program related to personal computers? Do you have an idea for developing such a project? If so, we would like to hear from you. The Brady Communications Company produces a complete range of books and applications software for the personal computer market. We invite you to write to David Culverwell, Publishing Director, Brady Communications Company, Inc., Bowie, MD 20715.

Registered Trademarks

Limits of Liability and Disclaimer of Warranty

ACKNOWLEDGMENTS

The following people from Apple Computer, Inc. provided extraordinary assistance:

Clifford J. Schaffer did a thorough technical review of the book, which owes a great deal to his assistance and unflagging attention to saying it all in a technically correct way. He was an ideal reviewer because he cared enough to carefully verify the technical accuracy as well as the clarity of exposition of the book.

Bill Grimm reviewed the manuscript and pointed us toward the facts and ways of conveying some very sticky technical points to inexperienced computer users.

Peter Baumb shared with us his deep technical knowledge of the Apple //c.

Guillermo Ortiz dug out many of the points that were not in the books or readily apparent.

Linda Glish and Mark Goldstein gave us the support and assistance necessary to make the book a reality.

The following people from Gary Phillips & Associates and Brady Communications Company contributed to the production of this book:

Jacquelyn Smith managed the production of the manuscript and solved the numerous technical problems involved in completing the book. She also supplied some of the photographs used in the book.

Michael Rogers of Brady Communications Company did a superb job of copy editing and designing the book.

Chris Williams of Brady Communications Company helped us to conceive the book and solve the various problems that arose on the way to completion.

Robert Arbegast did an excellent job of preparing the manuscript and editing many of the chapters.

The following people also contributed to the completion of this book:

Michael Fischer and Allan Woods provided a thoughtful and thorough review with helpful corrections and comments to ensure that this book is technically and grammatically correct.

Donald J. Scellato wishes to express his appreciation to Pauline, Matt, and Debbie Scellato for putting up with the late nights, strange rattling and beeping noises from the room that held the computers in the house when we worked on this project; for put-

ting up with the incessant clicking of keys and the buzzing of dot-matrix printers, and for being patient while the work went on.

Gary Phillips and Donald J. Scellato both wish to thank Apple Computer, Inc. for starting, maintaining, and steadily supporting the Apple II family of computers.

Some of the material in this book is based on *How To Select Software*, Copyright 1984 Gary Phillips & Associates or from *How To Select Hardware*, Copyright 1984 Gary Phillips & Associates.

INTRODUCTION

Congratulations! If you are reading this book you have purchased an Apple //c computer or you are seriously considering doing so. This book introduces you to the Apple //c and all of the things that can be done with it. This book is designed to be used with the manuals and disk tutorials supplied with the Apple //c. We have provided you with a wider range of topics, more in-depth coverage of many topics, and in some cases an alternative explanation of instructions reflecting our personal understanding of, and experiences with, personal computers. We feel that these chapters will be a valuable addition to the materials supplied by Apple with the //c.

The first part of the book deals with the computer system itself and how it operates. The first part also introduces you to advanced operations, alternative operating systems, and Applesoft BASIC. The second part provides a guide to selecting software and the various types of software available. The third part of the book deals with the "hardware" devices that support and extend the capability of the Apple //c computer. Each chapter is intended to stand alone in explaining a particular aspect of using the Apple //c. This was done to provide specific assistance to the person who has had previous experience with computers in general or with Apple computers.

Computer Systems

As you unpack your Apple //c, one of the first objects you unpack will be a rectangular box with a built-in keyboard. This is the basic computer. It contains a power supply, a printed circuit board that contains the computer's electronic memory, the keyboard, a disk drive that looks like a horizontal slot on the right hand side of the computer case, and a variety of plugs or sockets on the back of the case. By itself, the basic computer cannot communicate meaningfully with human beings. One other ingredient is required before you can carry on a meaningful and useful dialog with the computer. The missing device is some sort of video display. The Apple //c can use a variety of devices to display images and text. These devices are an ordinary TV set, a special TV set called a monitor, or a special flat screen to give the Apple //c added portability.

These two elements, the computer and the video display, provide the experienced programmer with nearly everything needed to use the system. However, most of the computer's memory is volatile. It

loses its contents each time the computer's power is turned off. The Apple //c, like most other computer systems, uses a device to store memory contents such as programs or data files. This device is called a floppy disk drive. The slot on the right hand side of the Apple //c is the slot where a magnetic disk is inserted. This flat, magnetic disk is called a floppy disk due to its flexibility. It is made of the same type of material used to make recording tape used in cassettes. The disk drive can read data from a floppy disk or can write data onto the disk. In the state just described, the computer system is still not useful to anyone but a programmer and the programmer will not be able to make the computer communicate with the disk drive.

The ingredient that makes this communication and the management of stored data possible is called an operating system. In the case of the Apple //c, the operating system must be loaded into the computer's memory from a floppy disk. The standard operating system for the Apple //c is ProDOS. This stands for "Professional Disk Operating System."

ProDOS is a powerful, modern disk operating system. Its functions are available in four distinct formats. All Apple //c users need to be familiar with the supplied *System Utilities* disk. It contains ProDOS plus a simplified menu system named STARTUP. The menu system makes most of the features of ProDOS available in a very easy-to-use way, through selecting functions from menus and answering questions.

The *System Utilities* go beyond ProDOS and provide many functions useful in dealing with disks and files for two other major operating systems available for the Apple //c: DOS 3.3 and Pascal, as well as the older DOS 3.2.

These alternative operating systems will be compared and discussed in the second part of the book. The objects described so far, with the exception of the operating system, make up the part of a computer system known as the "hardware." Hardware are the physical devices that can be handled. "System hardware" consists of all the devices that are intrinsic to the system's operation.

In the case of the computer system described above, the devices needed to make it operate are the computer, which is made up of a circuit board containing a central processing chip, memory chips, and chips to support system operation; a keyboard or other input device such as a mouse allowing humans to direct the operation of the system, and a video display device allowing the computer to communicate with human beings; and a mass data storage device such as a disk drive that provides a safe place to store programs, text files, and other data. A computer system may also have additional hardware that extends its capabilities beyond the basic requirements of the system. This additional hardware is known as the system's per-

ipherals. Peripherals consist of devices such as printers, plotters, modems, graphics tablets, and similar devices. The third part of this book provides detailed guidance on how peripherals work and how to select them for your Apple //c.

Software

As mentioned earlier, the physical elements of a computer system are not worth very much to any user unless an operating system and other programs known as applications software are available for use by the system. The operating systems for the Apple //c were mentioned earlier. Any program that must be used by a computer system to carry out its operation is called "systems software." ProDOS allows the computer to operate its mass storage devices. Any program that allows a computer to carry out a useful task for the computer's operator is called "applications software." There are many such programs available for the Apple //c. They include word processors, program writing utilities, graphics utilities, business graphics, data management programs, educational programs, home management programs, electronic spreadsheets, telecommunications programs, and countless others. All such programs must be written in a fashion that employs the computer's operating system, or in a way that uses the language employed by the computer to execute the user's instructions.

The Apple //c understands an electronic code called machine language. A number of methods are used to translate human instructions to machine language. The Apple //c has an Applesoft BASIC interpreter built into a small, permanent portion of its memory. This portion of memory is known as ROM, which stands for "Read Only Memory." ROM is not erased when the computer's power is turned off. The portion of memory described earlier as being volatile is called RAM. RAM stands for "Random Access Memory." The language built into the Apple //c's ROM is Applesoft BASIC. This language consists of a number of human words that the interpreter converts to machine language allowing the computer to carry out tasks directed by the user. It is a very simple but powerful language that meets many programmer's needs. Applesoft BASIC is known as the Apple //c's resident language. Other languages may be loaded into the Apple //c's memory and used by the computer. Each such language is made up of very specific words or symbols that convert human instructions into instructions that the computer can understand. Applesoft BASIC is discussed in more detail in Chapter 8.

Now that you have read this general introduction to computer systems, you should be ready to learn about setting up an Apple //c and

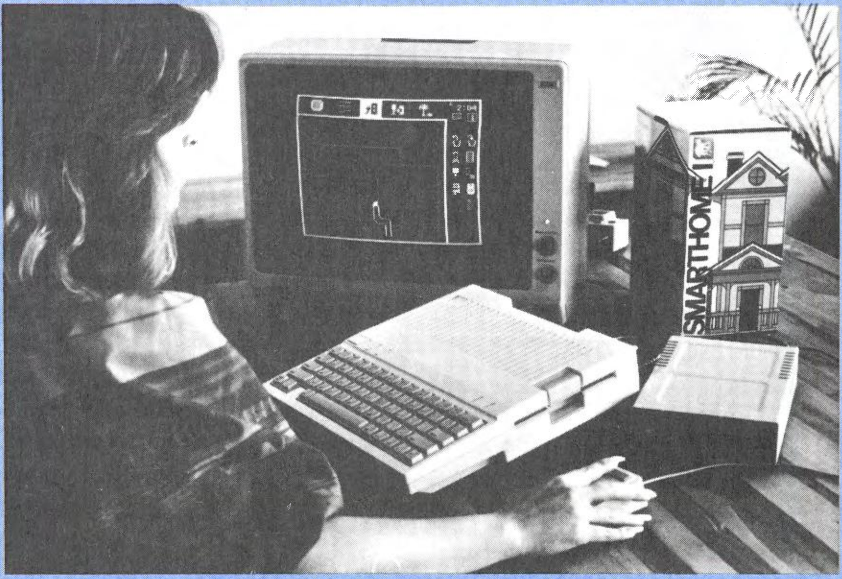
using it to accomplish the tasks that convinced you to purchase the computer. If you encounter any unexplained or bewildering terms in subsequent sections of the book, browse through those chapters that deal with the material in greater depth and their glossaries.

Apple //c Explorer's Disk, a diskette containing the sample programs used in this book plus eight additional programs, is available as an option to accompany this book. The additional programs include a ProDOS file display utility to show the contents of a ProDOS file sector by sector. There are also two indispensable programs to set up Imagewriter and Scribe printers for special types of printing. The remaining programs are Applesoft BASIC examples for further study and enjoyment for those who want to explore more deeply into Applesoft. Used with Appendix D (Applesoft Commands) and the Applesoft BASIC and ProDOS Reference Card, these sample programs will open the doorway to all of the features of Applesoft BASIC.

The Apple //c

1

Why the Apple //c Computer?



BACKGROUND

Many Apple users take pride in being part of the "Apple phenomenon." Over a million people use Apple computers nationwide. Many participate in Apple Users Groups (see Appendix B for the one nearest you). The Apple computer often plays a large role in family life or in the day-to-day workings of a small business. The Apple //c is the latest addition to a long and distinguished line of Apple computers.

Since Apple introduced its first "personal computer" in 1978, it seems that every year has seen the introduction of new computers by Apple and other manufacturers. A number of personal computers, home computers, or microcomputers–as they are frequently called–have been introduced, lasted a short time, and then ceased to exist because of competition, market conditions, the introduction of more powerful computers, or even because of bad company management. The Japanese have tried to invade the American market with microcomputer products with little success. There is a flourishing microcomputer industry in England, but except for the low-end Timex-Sinclair it has not effectively spread its products to the United States. The American microcomputer market has changed since 1978, as the microcomputer industry has matured. The products have evolved not only with the introduction of new technology, which has brought down prices and increased the capabilities of microcomputers, but they have changed to meet the specific demands and needs of microcomputer buyers. One of the primary winners in the change and turmoil of the microcomputer market is Apple Computer, Inc., the manufacturer of the Apple //c.

When the Apple II computer was introduced in 1978, there were few competitors trying to sell small computers. Computers were generally found only in business and government. Few people could afford the big and medium-sized units that were generally available. Radio Shack (Tandy) had a small computer on the market. Commodore also had one. IMSAI, Ohio Scientific, Atari, Texas Instruments, and other computer manufacturers also entered the market.

The computers introduced by these manufacturers had particular characteristics that set each apart from the rest. Each of these computers was an extremely powerful tool for any individual who had the money to buy one and the time to learn how to use it. Radio Shack had a tendency to introduce a new computer that did not operate quite the same way as its predecessors. Atari built computers that were extremely powerful, but did not provide convenient means of loading programs. In the first few years, programs and data were

The Apple //c computer can become a focus for family sharing and discussion.

stored on cassette tapes. In 1979 and 1980, new storage devices called disk drives were introduced. They looked and ran like funny-looking record players, but functioned like tape recorders.

Apple computers set some important sales trends because they were well made and well supported by hardware and software manufacturers. They were computers that worked well and could do anything because of their built-in, user accessible expansion slots. By 1983, when the Apple //e and Lisa were introduced, more than one million Apple computers had been sold all over the world. These

sales were made in the face of IBM's entry into the microcomputer field in late 1981, and in the face of other competition.

TODAY'S ENVIRONMENT AND THE APPLE //c

The Apple //c was introduced on April 24, 1984, in San Francisco, at an event called Apple II Forever. The timing of the introduction of this new computer by Apple Computer, Inc. has a number of obvious messages. Some of these messages are associated with Apple's overall business philosophy. This is reflected by the computer itself, how it is made, and what it can do. Some of these messages are associated with Apple's perception of the microcomputer marketplace and Apple's place in it.

With the announcement of the Macintosh in January of 1984, Apple Computer declared that the company was not going to lose itself in the crowd of companies pushing to produce IBM compatible machines. Instead, Apple re-established its position as an innovator in the microcomputer business. The announcement of the Apple //c in April of 1984, not only reaffirms the January declaration of independence and innovation, but makes firm Apple's commitment to the Apple II family of computers. It is a declaration to hardware and software manufacturers that Apple is not abandoning its past efforts in favor of a completely new computer with radically new and different operating characteristics. Instead, Apple is offering the public not only a computer similar to those already in place, but also a computer that offers more power and more convenience than earlier versions of the Apple II line.

Software manufacturers have two choices in approaching the Apple //c. They may continue to offer programs developed for earlier Apples. These will in nearly all cases work fine on the //c. Or, they may choose to develop new programs that take advantage of the //c's standard operating system (called ProDOS), 128K of built-in memory, mouse technology, and a heavier commitment to serial interfacing. However, the changes required to take advantage of these changes are very small compared to an investment to support a radically altered machine such as Apple's competitors tend to offer.

The Apple //c is specifically designed for families worldwide. Apple //c's are made with keyboards and character sets for many foreign languages. The back panel and many of the accessory connecters use icons that require no language. It is a computer with the

general characteristics of the Apple //e computer packaged in a transportable fashion. The Apple //c is aimed at the home market environment and the new, inexperienced computer user rather than at the more experienced "hacker" or "tinkerer." The Apple //c differs from the earlier members of the Apple II family in a significant way. The Apple //c is a self-contained computer that can be plugged into a home TV set or other video display device, and it can use the bulk of programs currently available for the other members of the Apple II family. It has all of the operating features of a slightly expanded Apple //e without requiring the user to invest in additional hardware for extending the computer's capabilities or memory capacity.

All of the earlier members of the Apple II family required the purchaser to make additional investment for expanding memory, communications circuitry, video expansion circuitry, and disk drives. The only additional investment for most of the new Apple //c owners, to realize the full potential of this new computer, is the purchase of a printer. All other requirements are built into the computer. Additional hardware such as an external disk drive, a dedicated portable or non-portable video monitor, a modem for telecommunications, and a number of other devices are available to enhance the capabilities of the computer beyond the basic machine. None of these devices (except perhaps the printer), is required to put the computer into full productivity.

WHAT IS AN APPLE //c COMPUTER?

Now that you know a little about the background of the Apple //c and why it exists, it is time to learn about what it really is. The Apple //c is equipped with a more powerful version of the microprocessing unit chip that operates the other members of the Apple II family. This chip is called a 65C02 chip. This chip contains the circuitry required to operate the Apple //c, to control communication with the Apple //c's dependent devices, and to address all of the electronic memory resident in the Apple //c. The Apple //c is equipped with 128K of random access memory (RAM). This is the working memory of the computer that holds the programs you use while they run.

RAM is the ultra-fast scratchpad memory of the Apple //c. RAM loses its contents when you turn off the power to the computer, so anything you will need again must be stored using the built-in disk drive. This memory also contains the computer's video display buffer where space is dedicated to define what should be displayed

on the screen. Data can be written or copied into RAM while it is being used and can be replaced with new data as required.

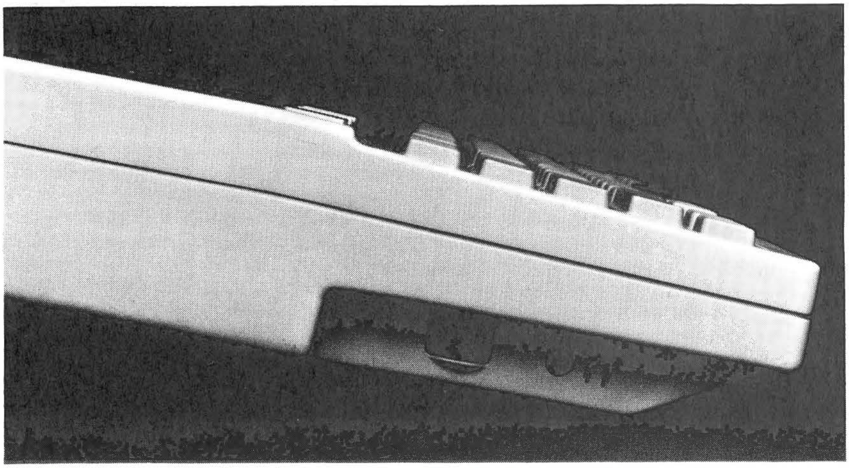
The Apple //c is equipped with 12K of read only memory (ROM). This is permanent memory that contains the interpreter for Applesoft BASIC, the native computer language of the Apple II family. (This permanent memory also contains a tool for examining machine language programs, called a disassembler, and the machine language routines that control the interpretation of data from the keyboard and from other sources.) An additional 4K of ROM controls communications with serial output devices such as printers, and serial input/output devices such as modems (communications devices that allow contact with other computers, teletypes, and so forth, either directly or via telephone lines).

The Apple //c has a built-in disk drive for mass data storage and retrieval. It has a built-in 80-column video display as well as a 40-column display mode for use with home TVs. The 80-column or 40-column mode is set by a switch built into the computer case. This switch can be read by a program designed for either a 40- or 80-column display to determine which the user wants. It is equipped with a high-quality U.S.-style 63-key keyboard allowing the user to communicate data and commands to the computer. This keyboard may be switched at the user's discretion to a keyboard called the Dvorak keyboard, preferred by some educators. The U.S.-style keyboard has four directional arrow keys, a shift key, a caps lock key, two Apple keys for extra functions, a tab key, and a delete key. The keyboard has full travel keys with both tactile and non-electronic auditory feedback for key presses.

The computer has a built-in speaker with volume control, and an earphone jack. The speaker is disconnected when the earphone is plugged in. The Apple //c has internal circuitry to support a serial printer via one of the external ports on the back of the computer. It also has a second serial interface circuit to connect with a modem for telephone or other data communications via an external port on the back of the computer.

The Apple //c can produce low-resolution color graphics displays, high-resolution color graphics displays, and double-high-resolution graphics displays. The low-resolution display is 40 horizontal by 40 vertical characters in 16 colors. The high-resolution graphics display is 280 horizontal by 192 vertical dots (or pixels). It will support six colors. The double-high-resolution graphics display is 560 horizontal by 192 vertical pixels with 16 colors.

The Apple //c has two different character (letter and number) sets available for use. The primary character set may be displayed in normal (white on black), inverse (black on white), and flashing modes. The alternate character set cannot use the flashing mode. The char-



The Apple //c has a speaker built into the bottom. It also has an earphone jack for private enjoyment. Notice the volume control, which works with the speaker or the earphone.

acter sets are stored in a single Read Only Memory(ROM) chip. This chip can also generate a special character set called MouseText. MouseText characters are generated by using a special command sequence. The graphics characters generated in this fashion may be moved anywhere on the screen with the mouse.

All of the things mentioned above boil down to this: the Apple //c is powerful. This computer is very compact yet very sturdy in design. It weighs about seven pounds and has a carrying handle on the back of the computer case. Power is provided by a compact external power supply that plugs into a built-in voltage converter.

As mentioned earlier, the Apple //c is packaged with the first-time computer user in mind. It is supplied with an extremely easy to understand *User's Manual* and a number of instructional disks that include actual programs to give the new user some insight into the many things the Apple //c can accomplish.

WHAT SETS THE APPLE //c APART?

The Apple //c is a compact, transportable computer. It is self-contained with a built-in disk drive controlled by a single high-capacity chip. It contains internal circuits to take the place of plug-in cards required by the Apple //e and the other Apple II computers for printer control, expanded video display, expanded memory, and

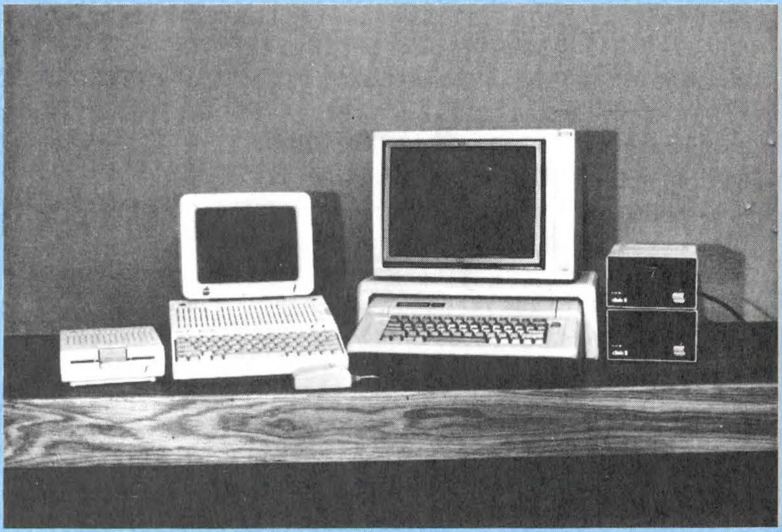
telecomputing. It supports either the 40-column or 80-column screen display. It supports black and white, standard NTSC color, and color video output which may be converted by a plug-in adapter for RGB display. It comes with a radio frequency modulator for connecting the computer directly to a standard TV set.

Besides the physical characteristics and electronic characteristics described above, the Apple //c is capable of running nearly all of the software (programs) released for the Apple II+ and Apple //e computers. It can be used for word processing, business applications, computer communications, exceptional graphics, educational purposes, budgeting, and games.

The Apple //c will be introduced to you in greater detail as you read this book. The computer's operation and set-up is described, and advice is provided on selecting hardware and software. By the time you finish this book you will be completely familiar with the Apple //c, and its use, and will be prepared to make informed decisions on buying software and expansion hardware.

2

The Apple //c Computer and the Apple II Family



One of the most attractive characteristics of the Apple //c is its compatibility with other members of the Apple II family. The term compatible, applied to a computer system, means that it can accept hardware and software that was designed for another computer system. If a system can use hardware or software that was used by a system introduced at an earlier time, it is said to be downward compatible with that system. The Apple //c is downward compatible in its use of most software with the Apple II+ and the Apple //e computers. It is downward compatible in its use of many hardware or peripheral devices that are not actual plug-in expansion card devices. It is not downward compatible in the use of parallel-interfaced printers. It is not necessarily plug compatible with all such devices. This means that the actual plug connecting the device to the Apple //c may not fit the sockets on the back of the computer. We expect to see adapters for common accessories supplied by either the manufacturers of the devices or from special cable makers. Manufacturers are already introducing serial to parallel converters for use with parallel printers.

The best way to expand your Apple //c is to stick to devices that have been specifically made or cabled for the Apple //c, either by Apple or by other vendors. Apple has a relatively complete line of accessories for the //c, all completely compatible and styled to match the //c's attractive appearance.

PRINTERS

The Apple Imagewriter printer can be connected directly to the Apple //c with a connecting cord supplied by your authorized dealer when you purchase the computer and the printer. This printer is an excellent dot-matrix printer that produces both graphics and text output. The text printed by the Imagewriter is very close in quality to that of letter-quality printers.

The Apple Scribe printer, a new portable printer recently introduced by Apple Computer, Inc., can be used with the Apple //c. Scribe combines the properties of the thermal printer with those of a dot-matrix printer to produce excellent quality output. It is the first printer that thermally transfers material from a ribbon onto ordinary paper. It can do dot-matrix style graphics and even do four colors with a special color ribbon. This is a truly revolutionary printer that deserves consideration as your Apple //c printer.

Both the Imagewriter and Scribe printers require special codes to use all of their functions. The disk, *Apple //c Explorer's Disk* (avail-

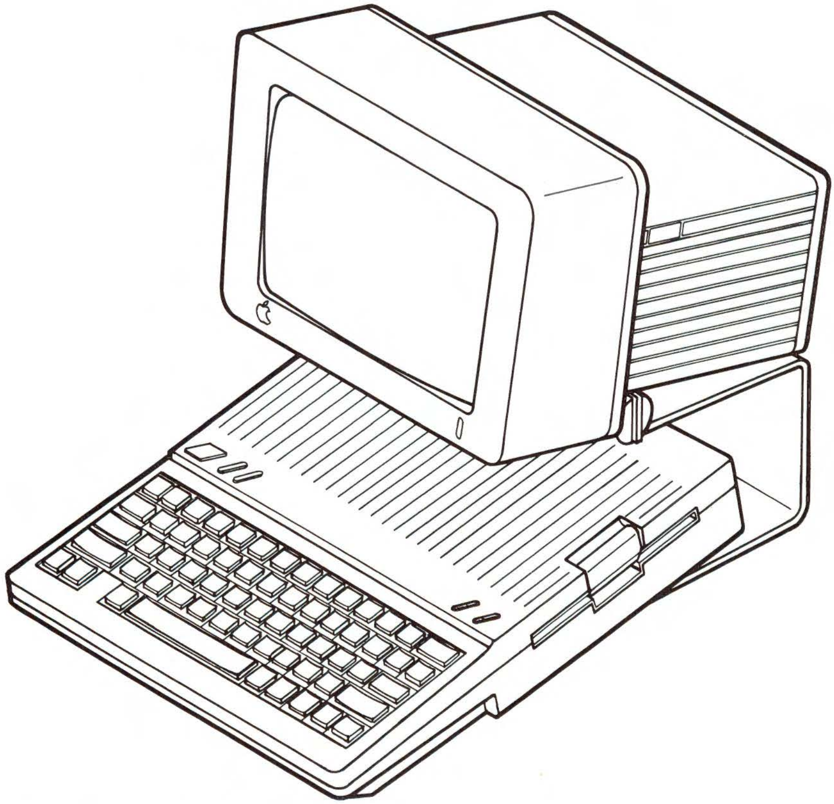
able for use with this book), contains programs to control the features of these printers through easy-to-use menus.

Many other serial interface printers may be used by the Apple //c if the proper plugs or connector cables are obtained. Be sure to consult the documentation or other instructions supplied by the manufacturer for such things as switch setting, rate of data transfer, and communications procedures between the computer and the printer before making the purchase. Be sure to consult the dealer if this other material is incomplete or does not answer your questions regarding how to connect the device to the Apple //c. If necessary, contact the printer manufacturer to obtain information that is not available from these preliminary sources. If no other information is available, and information is available pertaining to the Apple //e with Apple's Super Serial Card installed, follow the procedures outlined for the Super Serial Card as the Apple //c uses a subset of Super Serial Card protocols. Printers are covered in more detail in Chapter 13.

MONITORS

Monitors are special high-quality video display devices resembling TV sets but they produce a better display quality than a TV set. If you intend to use your computer for word processing, or for any 80-column use such as spreadsheets, a monitor is required. This is so because the monitor's higher resolution (crispness, definition, and steadiness) of the characters you see on the screen is required for the smaller 80-column characters, while the resolution of a black and white or color TV set is not adequate for 80 columns. Monitors come in a variety of types. They may produce black and white displays, green on black displays, amber on black displays, and color displays. Monochrome monitors (B/W, green, and amber) and composite color monitors connect to the Apple //c via the video port in the center of the back of the Apple //c.

Color monitors are available in two types. The first type is called a composite color monitor. The second type is called RGB for its three separate inputs: red, green, and blue. The RGB monitor produces a display superior to the composite color monitor. Composite color devices may be attached to the Apple //c via a cable connected to the third port from the left as you face the back panel of the Apple //c. This connection is marked with an icon that looks like a video screen with parallel lines, one on either side of the screen. The cables supplied by Apple Computer, Inc. are marked with the same icon. An icon is a simple drawing to replace a word or sentence to explain something. Monitors are usually not computer specific. This means that they will operate with any computer.



The Apple //c with the Apple //c Monitor and stand.

There are two monitors produced specifically for the Apple //c. One of these is a small, thin, rectangular flat screen monitor, which is portable. It sits comfortably on top of the Apple //c's case. This revolutionary monitor, with its extraordinary portability, will not be available until some months after the release of the Apple //c. The second is the Monitor //c, which matches the color of the Apple //c. It rests on a special stand to provide proper eye height for the user and it allows the Apple //c to be propped up by its carrying case handle for comfort and ventilation.

The selection of monitors is dictated by how you intend to use them. If you intend to play games and use programs with the 40-column display, your black and white TV set is adequate. If you intend to use games with 40-column programs that require color, your color TV set will probably be adequate although some color TVs will not do an acceptable job with computer-generated signals. If

you need an 80-column display for word processing or spreadsheet applications, which generate relatively small print, the black and white, green, or amber monitors are necessary for the job. If you need color, the quality of the graphics display that you require dictates whether you choose a composite color monitor or an RGB color monitor. Monitors are discussed more thoroughly in Chapter 12.

DISK DRIVES

Although the Apple //c has a built-in disk drive, there are occasions when a second disk drive will make the use of your Apple //c more convenient and more efficient. Two occasions making a second disk drive very desirable are when you are copying disks or files and when you are running programs that tend to tie up the internal disk drive. In the first case, the second disk drive saves the time involved in swapping the original disk and the copy disk in and out of the disk drive. In the second case, the internal disk drive is tied up by the program and a second disk drive is more convenient for storing and retrieving data files generated by the program.

The Apple Disk //c is a slimline, additional external disk drive designed especially for the Apple //c. It is compatible with all Apple hardware and software for the //c. It is also coordinated in color and styling to match the Apple //c. Its characteristics are exactly like the built-in disk drive of the Apple //c.

There are many other disk drives available for the Apple II, the Apple II+, and the Apple //e. Most of these disk drives, which are 5 1/4-inch disk drives with 35-track capacity, can operate with the Apple //c if the proper connector is installed with DOS 3.2, DOS 3.3, ProDOS, or Pascal. We expect that many of these will be certified for use with the Apple //c and provided with appropriate cables. Adapter cables may also be available. There is a possibility that drives not specifically certified for use with the Apple //c may cause problems and may violate FCC regulations on emission of radio frequency noise (which can interfere with radio and TV reception or interfere with the operation of your Apple //c or other electronic devices). Disk drives are explained in more detail in Chapter 14.

MODEMS

Modems are devices allowing your computer to communicate with other computers directly or over telephone lines. The Apple //c has the circuitry and port to connect the computer directly to a modem.

The program disk provided with most modems contains routines for setting up communications protocols, dialing numbers, connecting to other computers, transferring files, and other tasks required for successful communications via computer. The protocols that can be set are related to the rate of data transfer, the signals which tell one computer when the other computer is ready to send or receive data, whether the keyboard signals are repeated on the screen, and other controlling factors called "protocols."

Apple Computer, Inc. offers two modems that will operate with the Apple //c. Apple also offers an excellent communications program called Apple Access II. Before you purchase a modem you should consider the baud rate (rate of data transfer). This rate, usually 110, 300, or 1,200, determines how long you will be connected with the other computer, running up your phone bill and connect-time charges from on-line services while you are transferring data. When you consider modems, you should also consider the terminal or communications program that best satisfies your needs. This program can be a very simple program that dials and connects you. It can be a program that makes some other type of computer think that your Apple //c is the same type of computer. This type of program is called a terminal emulator program. It can also be a sophisticated data transfer program that uses single key presses to instruct the computer to dial a number and give the other computer your sign on code and any other codes that may be required for you to communicate. Such single key commands are called "macros."

The modems used by the Apple II, the Apple II+, and the Apple //e require connections through a plug-in card or they are modems that are built into the card. Apple Computer, Inc. sells two modems that can be used by the Apple //c. These are the Apple Modem 300 and the Apple Modem 1200. If you decide to purchase a modem that is not specifically certified for use on the Apple //c, be sure to examine the manner in which the modem is connected, and any features that are controlled by the plug-in card that may not be available for the Apple //c's internal circuits. The *Apple //c Technical Reference Manual* discusses all of the information you require. This information should be reviewed along with the manual supplied with the modem. Before you purchase a modem, make sure it is compatible with the Apple //c, and that the manufacturer has provided instructions for connecting it to the Apple //c. Modems are covered in more detail in Chapter 15.

PROGRAMMING LANGUAGES

The Apple //c has Applesoft BASIC resident in ROM. This language is also resident in the ROM of the Apple II+ and the Apple

//e. Applesoft BASIC is used for many of the programs written for the Apple II family of computers. The Apple //c also supports Pascal, an extremely powerful language used for many database applications, business programs, management programs, and similar applications. This language is loaded into the upper 16K of the 64K RAM built into the Apple //c.

FORTRAN, a science/business language, and SuperPilot, an educational authoring language, can be loaded into RAM and used on the Apple //c. Logo, a language that teaches young children the principles of logical thought and programming, is also available. The Apple //c user can also program in 6502 assembly language, which relies on mnemonic instructions rather than reserved words such as those used by BASIC to give instructions to the computer. Other languages such as LISP, C, and FORTH are available from independent vendors.

OPERATING SYSTEMS

Three operating systems are available for use with the Apple //c. Operating systems allow the computer to transfer data files and programs back and forth between the computer's memory and the disk drive. The primary operating system is ProDOS, introduced by Apple Computer, Inc. in 1983. This operating system is a complete system with commands available either directly through machine language interface calls, or as operating system commands similar to DOS 3.3 commands from Applesoft BASIC. The Applesoft command interface is accomplished by the program BASIC.SYSTEM. ProDOS utility programs are also available to the user as a collection of menus that allow the selection of features, such as copying disks, copying files, and managing files.

The second operating system available for the Apple //c is DOS 3.3. DOS 3.3 is the standard operating system for the Apple II, the Apple II+, and the early Apple //e computers. It is an operating system employed by many programs that operate on the Apple //c. This operating system is command-driven, but it uses utility programs to copy disks, copy files, and manipulate files. The third operating system supported by the Apple //c is Pascal (the UCSD p-System). This is the file handling system for Pascal, FORTRAN, and Pilot programming languages.

In addition to the various operating systems that are available, the Apple //c comes with a powerful and convenient utility called the *System Utilities* disk. This disk provides some of the same services as the ProDOS utility programs in a way even more convenient than the

menus in the ProDOS utility programs on the *ProDOS User's Disk*. Additionally, many of the menu selections available on the *System Utilities* disk will work on either ProDOS, DOS 3.3, or Pascal disks.

SOFTWARE

One of the advantages of the Apple //c is that it can use much of the commercial and public domain software that runs on the Apple II+ and much of the software that runs on the Apple //e, but it is not dependent on a particular plug-in card or upon serial output to produce results. Applications written for the Apple //e that run on the Apple //c include personal finance applications, productivity tools, educational software, communication software, games and entertainment software, and much more. Some examples of programs that are available for the Apple //e and Apple //c are shown here.

Application	Product	Manufacturer
Personal finance	Dollars and Sense	Monogram
Personal finance	Financial Cookbook	Electronic Arts
Integrated system	Appleworks	Apple Computer, Inc.
Graphics (mouse)	MousePaint	Apple Computer, Inc.
General finance	Multiplan	Microsoft
Word processor	Bank Street Writer	Broderbund
Database management	pfs:File	Software Publishing
Database management	pfs:Report	Software Publishing
Education	Rocky's Boots	The Learning Company
Education	Mastertype	Scarborough Systems
Education	Mastering the SAT	CBS Software
Education	Apple Education Classic	Apple Computer, Inc.
Education	Crypto Cube	DesignWare
Education	Apple Logo II	Apple Computer, Inc.
Education	Stickybear Shapes	Xerox Education Publications
Education	Grandma's House	Spinnaker Software
Education	Fact and Fiction Toolkit	Scholastic Software
Investment	Dow Jones Investor's Workshop	Dow Jones Software
Income Tax	Personal Tax Planner 1983/84	Aardvark/MacGraw-Hill
Communications	Apple Access II	Apple Computer, Inc.
Entertainment	Chess!	Odesta
Entertainment	Zaxxon	Datasoft

The programs listed above are only examples of diverse programs that can operate on the Apple //c computer. Chapters 9 and 10 provide in-depth coverage of software types and guide you in selecting software.

MOUSE

The Apple //c has a port on the rear panel that can be used for a mouse or a specially designed joystick or game controller. One mouse that works via this port is the AppleMouse designed for the Apple II+, the Apple //e, the Apple //c , the Macintosh, and Lisa. This pointing device will be used by many software products. We expect to see an explosion of high-quality mouse-based software soon, as the mouse becomes more generally accepted than it is today. The AppleMouse does not require an interface card on the Apple //c. It plugs directly into the mouseport on the back of the computer. The mouse is extremely friendly and powerful when it is used. One example of an excellent mouse-based program is MousePaint. This program is a wonderful tool to assist in drawing pictures, diagrams, and other useful graphics. For more on the mouse, see Chapter 16.

SYSTEM COMPARISONS

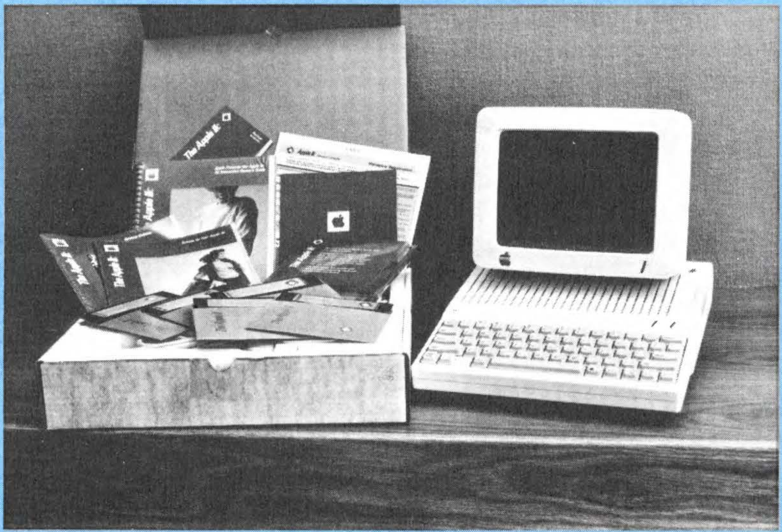
We have discussed in some detail the fact that the Apple //c is largely compatible with the other members of the Apple II family—the Apple II, the Apple II+, and the Apple //e. The following table compares these computers.

Characteristics	Apple II, II+	Apple //e	Apple //c
Random access memory	4K to 48K expandable to 64K	64K expandable to 128k	128K
Read Only Memory	12K	16K	16K
Applesoft in ROM	II—No, II+—Yes	Yes	Yes
Autostart ROM	II—No, II+—Yes	Yes	Yes
Monitor program in ROM	Yes	Yes	Yes
Processor	6502	6502A	65C02
Built-in self test	No	Yes	No

Characteristics	Apple II, II+	Apple //e	Apple //c
Runs at 1 mHz	Yes	Yes	Yes
Keyboard keys	52	63	63
Auto repeat keys	No	Yes	Yes
Open/closed Apple keys	No	Yes	Yes
Upper/lower case keyboard built-in	No	Yes	Yes
Expansion slots	Yes	Yes	No (most common expansion functions built-in)
Standard Apple II family slot conventions	Yes	Yes	Yes
Serial ports built-in	No	No	Yes (2)
80-column display slot	Yes	Yes	No (built-in)
80-column display built in	No	No	Yes
Port for mouse built-in	No	No	Yes
Plug-in mouse controller card	Yes	Yes	No (port built in)
Plug-in disk controller	Yes	Yes	No (built in)
1 chip internal disk controller	No	No	Yes
Built-in disk drive	No	No	Yes
More than 2 drives	Yes	Yes	No (1 built in, 1 external optional)
Profile hard disk support	II—No, II+—Yes (w/64K)	Yes	No
Uses Apple II and II+ joysticks	Yes	Yes	No
Uses Apple //e joystick	No	Yes	Yes
Built-in RGB circuits	No	No	Yes
RGB accessible from plug-in board	Yes	Yes	No (access built in)

3

Setting up the Apple //c: A Guided Tour



The Apple //c is an extremely easy computer system to set up and put to work. This compact, transportable computer is designed to make set-up easy for beginning computer users. The Apple //c consists of a lightweight rectangular case with a built-in disk drive, a keyboard, a keyboard switch, a 40-/80-column switch, an earphone jack, a power-on light, a disk-use light, a volume control, and seven connection sockets on the rear of the computer case. If you buy the basic computer, you will receive the computer described above, an external power supply, an RF modulator, a switch box, three instruction books, a video cable, five disks, and a warranty card. Be sure to check the contents of your Apple //c box against the shipping list contained in the book.

You also need a video display to let the computer communicate with you, and you may want a printer to produce hardcopy of output from the computer.

When you take the computer out of the box, set it on a table or desk with the keyboard up and facing you. Make sure the carrying handle is in the down position. This props up the rear of the computer and makes it more comfortable for you to use. It also allows the computer to be cooled properly by ventilating the bottom of the computer. Proper ventilation and cooling is absolutely essential to the proper operation of your Apple //c. **Please, always** prop it up on the handle to allow air cooling and flow. You do not drive your car without oil, so do not operate your fine new computer without propping it up on its handle—the only legitimate operating position.

The Apple //c comes with everything you need to start computing except for your home TV.

Before going further, look the computer over carefully to be sure you understand its external features and how they work.

Your Apple //c comes with an excellent booklet, *Setting Up Your Apple //c*. You should rely on it as your guide in setting up the computer. In this chapter we will provide supplemental information and emphasize the points we found especially confusing or potentially difficult. This chapter is a supplement to *Setting Up Your Apple //c* and should not be considered a replacement for it.

The most obvious thing about the computer at this stage is the keyboard. The keyboard allows you to communicate with the computer and give it instructions. The keyboard has 63 keys that produce all 128 ASCII characters. The booklet *Apple Presents the Apple //c—An Interactive User's Guide* and the accompanying disk tutorial *An Introduction* provide an outstanding introduction to the keyboard and its features. The keyboard has a space bar on the bottom, a row of numbers across the top, an open Apple key on the bottom row of the keyboard to the left of the space bar, and a closed (solid) Apple key to the right of the the space bar. There are four directional arrows on the right side of the bottom row of keys. There is a *caps lock* key on the left side of the bottom row of keys. A *control* key is on the left side of the third row of keys and a *return* key is on the right side of the fourth row of keys. An ESC key is located on the left side of the top row of keys. A DELETE key is located on the right hand side of this row of keys.

If you look just above the keyboard you will see a large diamond-shaped key on the left side. This is the RESET key. Next to the RESET key are two narrow switches. The first switch offers the user the option of a 40-column display with the switch down and an 80-column display with the switch up, only if the program was designed to "read" the switch and produce the desired display. A similar switch is located next to this switch. The second switch allows the interpretation of the keys on the keyboard to be changed. This offers the user a Dvorak keyboard instead of the standard U.S. keyboard. The Dvorak keyboard is used when the switch is down. The regular keyboard is available when the switch is up.

There are two lights on the right side above the keyboard. The left light is the disk-use light. This light glows when the internal disk drive is in use. The light on the right side of the disk-use light is the power-on light. This glows when the computer power switch is on and the power cable is connected. An internal 5¼-inch disk drive is located on the right side of the computer. There is an audio jack and a volume control knob on the left side of the computer.

Turn the computer around and look at the back panel. You will see seven sockets and a switch. This part of the computer contains the Apple //c's power switch and the means for connecting the Apple

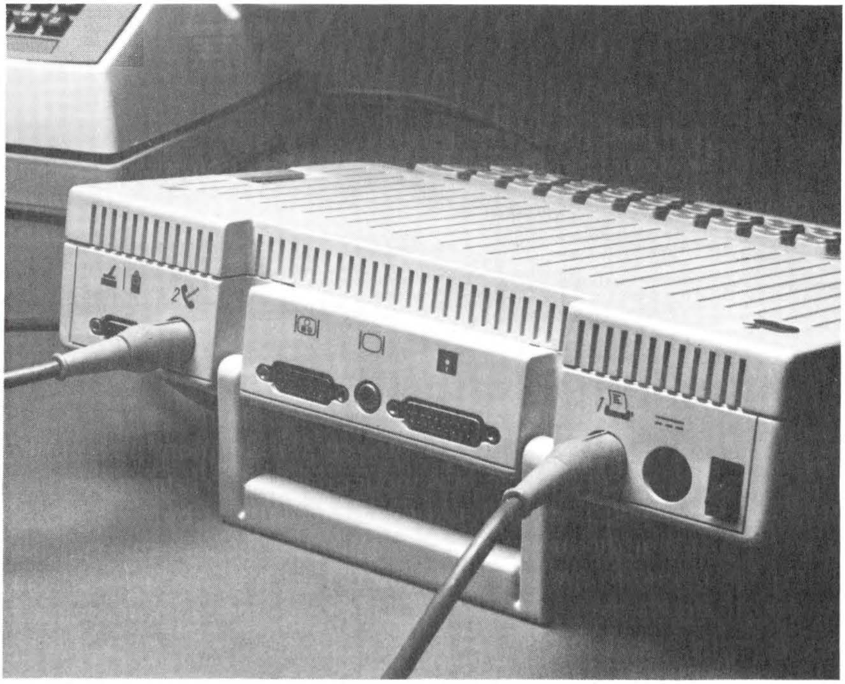
The Apple //c has a full-size, full-travel keyboard with all the standard keys and an excellent "feel" for touch typists.

//c to a video display monitor, a printer, another disk drive, a TV set, a modem, or a hand control mouse.

CONNECTING THE VIDEO DISPLAY

Now that you have examined the back panel of the Apple //c, connect your video display device to the computer. If you are using a TV set, the switch on the RF modulator should be set to channel 3 or 4. Choose the channel that is not used for TV broadcasting in your area. If both of these channels are used for broadcasts, set it to channel 3. Plug the RF modulator into the third socket (port) from the left. This port is marked with a TV set icon:

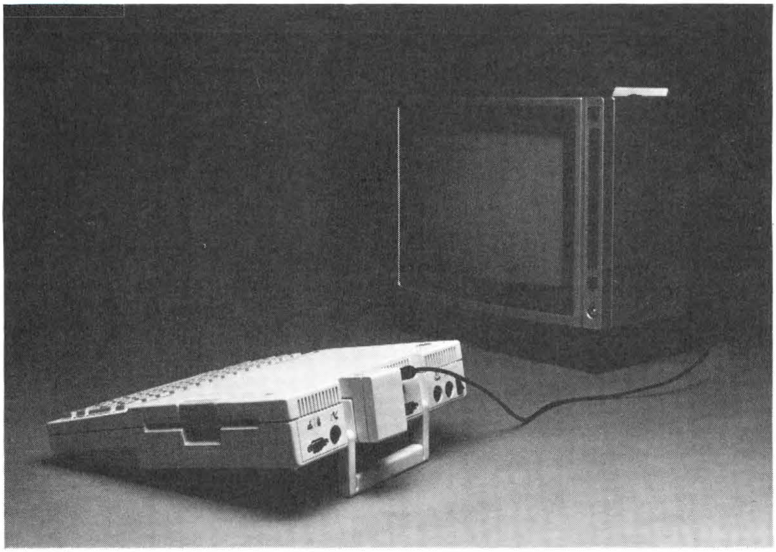
Set up your TV set by loosening the screws on the TV set marked VHF, detaching the two wires connected to those screws. Pick up the switch box and attach the antenna wires that were connected to the TV to the screws on the switch box. Now, tighten these screws.



The back of the Apple //c, illustrating the "ports" or connectors for accessories. Here a modem and serial printer are attached.

The switch box is a connector that allows you to slide a switch to settings marked "TV" or "Computer." The setting of this switch determines whether your TV set is used for showing TV broadcasts or displaying computer output. Slide the switch to the computer setting. Plug one end of the video cable into the switch box you have connected to your TV set. Plug the other end of this cable into the RF modulator that has been plugged into the proper socket on the back of the computer. Make sure the screws on either side of the RF modulator are tightened to provide a good, secure connection to the computer. Set the 80-/40-column switch on the left side of the computer (above the keyboard next to the *reset* switch) to the 40-column setting. This switch is in the 40-column setting when it is in the down position. Set the TV channel selector to channel 3 or 4 depending on the setting of the RF modulator, and plug the TV's power cord to a grounded wall socket.

You may have purchased a video display monitor such as the Apple //c monitor. It has a green display on a 9-inch screen and can be mounted on a special eye level stand. You may use the video cable supplied with the computer. You connect one end of this cable



Hooking up your TV to the Apple //c is easy. The final set-up is shown here.

to the center socket on the back of the computer. This socket is marked with the monitor icon:

Connect the other end of this cable to the socket on the monitor. Then, plug one end of the monitor's power cord into the back of the monitor and the other end into a three-hole wall socket. Make sure that the computer's power cord is similarly attached to the computer and an electrical outlet. Once all of these connections have been made, turn on the monitor. Insert a disk into the computer's built-in disk drive and turn on the computer's power switch.

If you plan on purchasing the convenient, portable flat panel display (available Fall 1984) you would connect it to the computer the same way you connected the monitor described above. If you have purchased a composite color monitor, you would connect its cable directly to the socket marked for the monitor without using the RF modulator. If you purchased an RGB monitor, connect it to an RGB adapter that can be plugged into the TV set socket.

EXTERNAL DISK DRIVES

An external disk drive draws its power from your Apple //c. The connection you make between the disk drive and the computer provides the means of transferring data to and from the computer's memory and also supplies power to the disk drive. The external disk drive is connected to the Apple //c via the port on the left center of the back panel marked with the disk drive icon:

The other end of this connector cable will already be attached to the disk drive by the manufacturer. Apple Computer, Inc. offers a half-high, $5\frac{1}{4}$-inch disk drive with the proper connecting cables and plugs for the Apple //c. A high-capacity "Hard Disk" is available for the //c. This would be connected to the disk drive socket.

PRINTERS AND PLOTTERS

Printers such as the Imagewriter and the Scribe must be connected to the right serial port marked with the printer icon:

The retaining screws for the computer end of the connector should be tightened when the connector is plugged into the socket on the back panel of the computer. The other end of this serial cable should be plugged into the back of the printer. Make sure the switch settings inside the printer are set properly for the Apple //c. Plug the printer's power cord into an electrical outlet. Turn on the printer. Then turn on the computer. Make sure the paper used by the printer has been loaded and the power indicator light on the printer is on. A plotter is connected in a similar fashion.

MODEMS

When you purchase a modem for use with the Apple //c it may have connection instructions different from those described here. These instructions apply to the Apple Modem 300 and the Apple Modem 1200. Make sure the Apple //c power switch and the modem power switch are off. Attach one end of the modem cable to the serial port on the left side of the Apple //c's back panel. This port is marked with the modem icon:

Attach the other end to the modem and tighten the retaining screws. Unplug your telephone cable from your telephone and attach it to one of the jacks on the back of the modem. Use the phone cable that was supplied with the modem to connect the telephone to the other jack on the back of the modem. Plug the small end of the modem power cable into the modem. This power cable is attached to a power transformer that is plugged into an electrical wall socket. Make any modem setting required by the instructions for the modem and you are ready to turn on the computer and load your terminal or communications program.

THE MOUSE OR HAND CONTROLLER

The mouse is used as a convenient pointing device for programs that employ it, and as a drawing device for MousePaint. The mouse is connected to the left nine-pin socket on the rear of the Apple //c. The icon for this socket is:

Hand controllers for games and for some graphics applications are connected to the same socket. All you need to do with either of these

devices is to plug it into the hand controller/mouse socket and tighten the retaining screws.

THE POWER SUPPLY

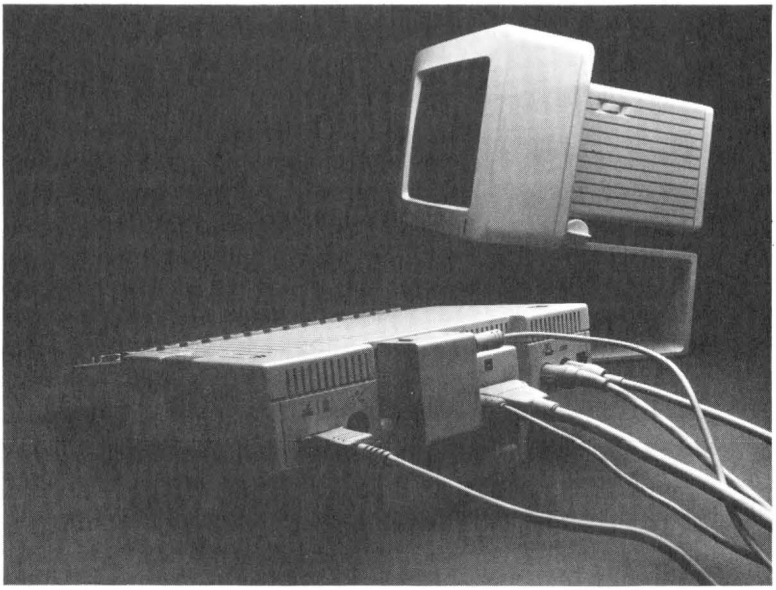
We have saved the description of the power supply connection until now to make a point. Power to the computer and to any peripheral being connected to it should be off while connections are being made. Once you are certain that a device is properly connected, the power may be turned on. The Apple //c's external power supply is connected to the socket on the right side of the back panel, next to the power switch. This socket is marked with the power icon:

The plug from the power supply has its top marked to ensure that it is inserted properly. Once the connection to the computer has been properly made, plug the power supply into an electrical socket and turn on the computer. The power indicator light above the keyboard will come on if everything is in order.

GUIDELINES FOR SETTING UP YOUR SYSTEM

You have purchased an extremely powerful and useful tool in the Apple //c. Its usefulness to you is extended by the external devices you employ with the computer. Each such device has the proper cables and plugs for its connection. Like the computer, each such device has instructions for connection and use. Read these instructions carefully. Follow them. If you have any questions about them, or if something does not work as explained in the manuals for the device, consult the dealer where you purchased it or consult the manufacturer.

Make sure the computer's power switch and the device's power switch are off while you are connecting or disconnecting the device and the computer. Attaching or disconnecting an accessory while the power is on could seriously damage your Apple //c, the accessory, or the cable. Make sure the device is connected to the proper port or socket on the back of the computer. Also be sure that all connections



This Apple //c is loaded. It is shown with everything it can handle attached except a modem. Notice that two video displays, both a monitor and a TV, are attached.

are made properly and tighten any retaining screws or retaining clips required by the connector. Arrange the parts of your system so they are convenient for you. Monitors should be at a comfortable level for your use, and not be sitting directly on the computer. The brightness and contrast controls should be at comfortable settings. Printers should be placed relatively close to the computer with paper and other supplies handy. The external disk drive should be placed at a convenient location for inserting and removing disks. If the cable that was supplied with a particular device for connection to the computer is too short or does not have the proper plug for connecting to the Apple //c, call your dealer for a replacement. If a replacement is not available, your dealer may be able to make one for you.

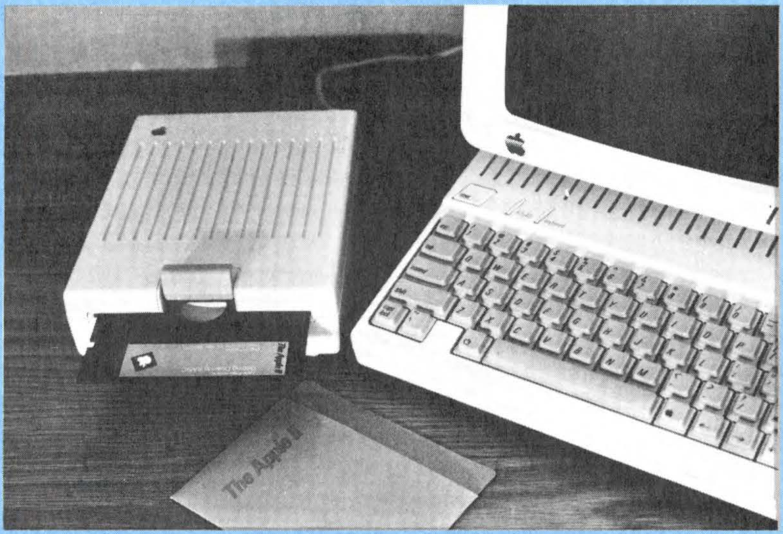
Above all, remember that your Apple //c needs a constant flow of cool air to avoid overheating and consequent damage or erratic functioning. Never operate the computer unless it is propped up on its handle and free of monitors, other devices, books, paper, or other obstructions. that could potentially block the flow of air. The Apple //c relies on quiet, energy-efficient natural flow of air to cool its electrical components. Please do not defeat this elegant and natural cooling system by blocking the air passages over, under, and around the

computer. Likewise, you should avoid smoking, eating, and drinking around the computer. Cigarette smoke and ashes, food crumbs, and liquids are harmful to the computer and especially to the disk drive and your disks.

If you follow these instructions for setting up your system, as well as those in *Setting Up Your Apple //c*, and follow the instructions that are supplied with any devices you connect to the Apple //c, you will be successful in using the system and will enjoy it for a long time.

4

Operating the Apple //c and Using the Disk Tutorials



Once your Apple //c is set up you are ready to put it through its paces. This is the fun part, and with the Apple //c and the excellent materials supplied with it you will have no trouble operating the machine. Apple provides you with an *Interactive User's Guide* with the Apple //c. The disk tutorials that accompany this manual provide instructions, examples of actual commercial programs available for the //c, and sample programs to make your first steps in using your computer as comfortable and as informative as possible. In this chapter we will try to provide you with some additional tips and hints to supplement the material in the *Interactive User's Guide*. We urge you to go through the guide first. If you encounter problems or things you do not understand while using the disks as directed in the *Interactive User's Guide*, perhaps our explanation of the disks and how to use them will clarify or provide a missing piece. A second run through these instructions should broaden and deepen your understanding. This chapter is not meant to replace this guide. Our purpose is to supplement and support that excellent guide.

Some of the disks supplied with the Apple //c have information written on both sides of the disk. Only one side is available at any given time. This will be the side that is up when you insert the disk. To access the other side you must remove the disk, turn it over, and put it back into the drive. This is exactly the way you use a standard phonograph album, playing one side of the record at a time. For more information on how disk drives work, see Chapter 14.

As you use the *Interactive User's Guide* and the tutorial disks, you will learn how to operate the Apple //c properly. You will gradually gain confidence in using the computer and learn new skills relevant to computing activities. At the same time you will be exposed to programs that are available for use on the Apple //c, programming languages such as Logo and BASIC, technical information about the computer and how it works, and the details of using the keyboard. When you have completed the tutorials, you will have most of the knowledge and skills needed to do a wide range of things with your Apple //c. You can always go back to the *Interactive User's Guide* and the disk tutorials if you need to brush up on these skills. The following observations should be helpful to you during your study of the disk tutorials and in your later use of your Apple //c.

TURNING ON THE SYSTEM

Before you flip on the power switch, it is a good idea to check all of the cords connecting the Apple//c to its peripherals to be sure they are securely inserted into their outlets at the back of the computer.

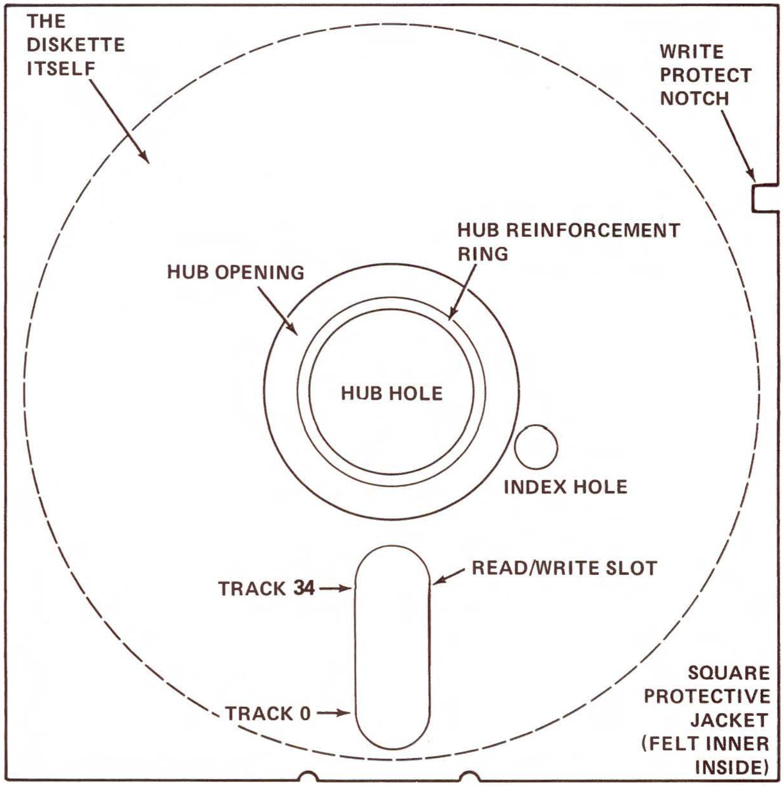
Choose the disk that you want to use for this session and insert it into the drive. Detailed instructions for handling and inserting disks follow. Turn on the monitor, TV, or other screen device, then turn on the computer with the power switch at the back left. You will hear a series of clicks as the disk drive reads the disk, then you will see a display on the screen that differs from disk to disk. When you begin regular use of the Apple //c, you will usually be seeing either the *System Utilities* main menu or the first display from the applications software you are using. There is very little that can go wrong in the power-up situation provided the proper disk is inserted correctly and the cables are attached correctly.

USING DISKS AND DISK DRIVES

Since disks are going to play a large role in your use of the Apple //c, you should understand a little about disks, the care of disks, and disk drives. We will go into more detail about how disk drives work in Chapter 14. Disks are a magnetic recording media similar to the tapes used in cassette recorders. The recording media is flat and circular, with a hole just over one inch in diameter in the center. This magnetic media is held in a slightly flexible cardboard or plastic sleeve that is coated with a protective material and lubricant on the inside. The hole in the center fits over a conical spindle or "hub" inside the disk drive. This centers the disk media accurately each time it is used. The sleeve has a finger-shaped cut-out that exposes the media and allows the disk drive's read/write head to come into contact with the media. This is the area where data is written onto or read back from the disk. Since the disk drive spins the disk around, all of the useful surface will pass under the open window for reading and writing of data.

Disk sleeves are customarily identified with a gummed paper label identifying the contents of the disk. It is very important to label disks adequately, since the information recorded on the brown mylar media is not visible to the eye. If you "lose" something on a disk by not labeling it properly, you may have to go through your entire collection of disks searching for the lost file using the *System Utilities* "Identify and Catalog a Disk" function. This will be very time-consuming, and depends on your having named the file in an easily recognizable way. The best plan is to always keep your disks clearly labeled. Boxes of blank disks usually contain blank gummed labels that conveniently fit on the top of the disk sleeve.

Labels for disks should be written before you stick them onto the disk sleeve whenever possible. If you must write onto a label on the



X-ray view of a diskette. (Do not x-ray your diskette—the data on it will be erased.)

disk, great care is needed because you can damage the disk and lose your data. When you write with a pencil or ballpoint pen on the top sheet of a pad of paper, an indented trace of what you wrote is left in the next few sheets. In the same way, if you write on a label attached to a disk jacket with a ballpoint pen or pencil, the disk media itself may easily be creased through the jacket. At best this can render the data unreadable, and you can easily imagine the problems a crease in the disk will cause as it runs past the internal read/write head of the disk drive. The only safe way to write on a label attached to a disk is with a soft-pointed instrument such as a felt-tip pen.

While there is very little you can do to damage the Apple //c (within the boundaries of good common sense), disks are quite fragile. They require careful and respectful treatment. They are subject to damage by temperature extremes, dust, dirt, spilled liquids, and

even fingerprints on the exposed part of the disk media that shows through the finger-shaped window. This is why disks are usually sold with a paper or plastic envelope that covers and protects the exposed part of the disk media. Disks should always be kept in their envelopes when not in use, even momentarily. It is just too easy to accidentally touch the exposed part of the disk through the window to take any chances with uncovered disks.

There are some other steps you should take to protect your disks. Keep them out of direct sunlight, since this can warp the jacket much like a record. Let the disks adjust to room temperature before you use them if they have been exposed to extreme cold or heat. They should be kept away from strong magnetic fields such as a toy magnet, an appliance or motor, or even the back of many monitors and TV sets. They should also be stored in an air-tight box. Many boxes are now available that help to organize your disks as well as to protect them from dust and other household hazards. Needless to say, disks also should not be bent, folded, stapled, or mutilated.

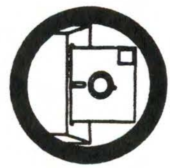
If this seems like much ado about nothing because of the low cost of disks, consider this. Many of the files and programs you will develop will represent an investment of tens or even hundreds of hours of your time. That is their real value. Once you are well on your way into the electronic age, you will accumulate a library of valuable commercial software and personal disks. To protect your financial and time investment in these disks, you will want to do two things. The first is to care for the disks as we have outlined. The second is to keep an extra or "backup" copy of each valuable disk. Then if something goes wrong with a disk, you can recover your files from the backup. In this case the very first thing you will do is to make a fresh backup, so you will not be caught without a spare. The simplest way to make a backup copy is through the *System Utilities* disk provided with the Apple //c. If the disk is copy-protected you may not be able to back it up. In this case you should contact the supplier for a backup copy.

There is a small notch on one or both edges of a disk used to allow writing to the disk. When the material recorded on the disk must be protected from erasure or from being written over, the notch is covered with a special piece of heavy tape supplied with the box of disks. This notch is called the write-protect notch and the short piece of heavy tape is called write-protect tape. Again, on the disks with an open notch, data can be changed, but on a disk with a closed notch, the data is protected. Unfortunately, the write-protect tab only protects the entire disk. If you have any files on the disk that must be changed, the disk cannot be kept in write-protect status. Backing up your disk is your ultimate protection from loss of valuable data and programs.

The internal disk drive of the Apple //c is a single-disk drive normally using single-sided disks with a write-protect notch on the left

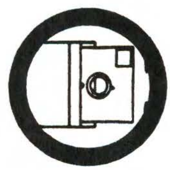
Keep these points in mind when using disks:

Precision surface.
No fingers, please!



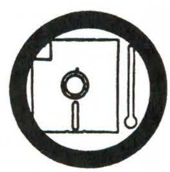
For your disk's sake
(and the system's, too)
insert disk carefully.

Magnetic fields erase.
Keep them far away.



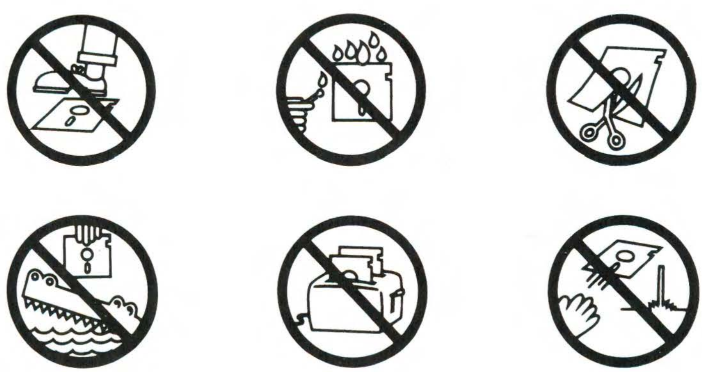
Keep it safe —
in the envelope
when not in use.

Bending and folding
may damage.
Handle with care.



Keep disks comfortable.
Store at: 10° to 52° C.
50° to 125° F.

Here are some more helpful hints on disk care:



(Courtesy of Bert Kersey and Beagle Bros.)

hand edge of the labeled side. A few of the disks provided with the Apple //c have two write-protect notches and data and programs on both sides. These are specially manufactured double-sided disks that must be flipped over to use the opposite side. The disk drive reads the bottom side of the disk and writes to that side. The bottom side of a disk is the side opposite the label.

When you insert a disk into the internal disk drive of the Apple //c, grasp the disk carefully with the label side up and the slot facing the disk drive slot. The disk drive door should be open (the catch is up) and there should be no other disk in the drive. Insert the disk carefully until it stops. Do not force or bend it. If it will not go all the way in, withdraw it carefully and try again. If it will not go in properly after several tries, take the computer to the dealer and have the drive mechanism checked. Do not try to open the computer case or try to repair or adjust the disk drive yourself or you will void the warranty of your Apple //c.

APPLE PRESENTS AN INTRODUCTION

If you are a new computer user or a first-time Apple //c user, you should select the *Apple Presents An Introduction* disk to learn about the Apple //c keyboard and the many ways to use it to communicate with your Apple //c computer. Take this disk from its protective jacket and insert it into the internal disk drive in the manner described above. Make sure your monitor or TV set is connected to the Apple //c and is turned on. Turn on the Apple //c's power switch. The disk drive will make some noise as it reads the program into the computer's memory. A graphics display will be printed on the video screen and a drawing of the keyboard will be displayed under the graphics. This display introduces the Apple //c. You will see a number of questions and instructions on the screen that will teach you how to use the keys on the Apple //c keyboard.

You will learn how to erase screen entries, insert material within entries that are already on the screen, how to capitalize letters and produce symbols, and how to maximize use of the Apple //c keyboard. The functions of special keys (such as the open and closed Apple keys) will be explained. When you have completed the exercises on this disk, you should have an understanding of the Apple //c's keyboard and how it functions, as well as the imaginative way programmers cause programs to interact with the computer user, respond to erroneous user responses, and prompt the user for responses. Think about the amount of advanced planning required to respond appropriately to anything that the student at the keyboard may do! The tutorial provided with the

Apple //c constitutes an excellent example of how computer programs should be written. They are state of the art programming and will serve you well as examples if you decide to go on to write your own programs. You are fortunate to have these excellent references. Apple is years ahead of most other computer companies in providing excellent tutorials.

APPLE AT PLAY

When you finish with the introduction to the Apple //c and its keyboard, you are ready to try the other tutorial disks that were supplied with your Apple //c. One of these disks is *Apple At Play.* Remove the introduction disk and place it back in its protective sleeve. This is a very important procedure if you wish to give the disks you use a long and useful life.

Insert the *Apple At Play* disk into the disk drive. The computer is still turned on, so you can try a procedure called a "cold boot" that makes the Apple //c think it has just been turned on. To do this press the open Apple key and the Control key together, then press the Reset key and let it up. Release the other two keys. This alters the computer's memory and instructs the computer to boot the disk that is currently in the default disk drive. The computer will beep and go through the same procedures discussed earlier in this chapter.

There is one possible problem in this procedure. If by chance you do not depress the open Apple key far enough to make contact, you will get a strange-looking screen with rows of inverse video "S's" across the screen near the top and bottom. If this happens, just try the cold boot again, being careful to depress all three keys (open Apple, control, and reset) completely. If all else fails turn the power off, wait 15 seconds, then turn the power back on.

Apple At Play contains six programs, an introductory program, and a routine to quit the disk. All of these options are listed on the menu. You may select an option from this menu by moving the menu indicator (angle brackets) around the list using the up and down arrow keys. See the list below.

```
The Apple at Play Menu
----------------------
1. <INTRODUCTION>
2. Apple 21
3. Financial Tools
4. Lemonade
```

```
5. Music Recital
6. Space Quarks
7. Quick Quiz
8. Quit
```

Press the RETURN key when the pointer rests on the program you wish to run. You may also enter the number of the program you want and press the RETURN key.

The programs on this disk are a combination of programs collected by Apple Computer, Inc., to introduce older versions of the Apple II family or newly gathered programs to demonstrate the Apple //c. One of the programs on the disk, Space Quarks, is a game that is available as a commercial product. Apple 21 is a high-resolution graphics version of the old favorite of poker players, blackjack or 21. Financial Tools is a selection of programs to compute interest charges, depreciation, or loan amounts. Lemonade is a game that teaches children the principles of business by simulating the operation of a lemonade stand. Music recital is an example of music that has been generated on the Apple //c. Space Quarks is a fast-paced, shoot-em-up type of arcade game. This game may be played under keyboard control or by using a game controller attached to the mouse/game controller port on the rear of the Apple //c. Quick Quiz is a game that tests your knowledge of the material in the *Interactive User's Guide* to the Apple //c. This disk effectively presents many of the Apple //c's graphics and sound generation capabilities. It also stresses the use of menus, input screens, program response to input, various styles of graphics output, and animation. This disk should give you an appreciation of how useful and entertaining the Apple //c can be.

APPLE AT WORK

The *Apple at Work* disk is a double-sided disk. This means that programs are stored on both sides of the disk. One side of the disk contains a 40-column version with a sample of a program called Appleworks. The other side contains the 80-column version of the same demonstration. The 40-column version is for those Apple //c owners who are using a TV set as their primary video display. If you are running the 40-column version of the *Apple at Work* disk, remember that Appleworks only operates in the 80-column environment. Appleworks is an integrated software application. It combines a database manager with an electronic spreadsheet and a word processor. It is a powerful and exciting program.

Apple at Work is not intended as just a demonstration of Appleworks. It is intended to demonstrate the capabilities of a word processor, a database manager, and an electronic spreadsheet. It uses demonstration screens with explanations to describe what is going on in the program, and requires some input from the user to demonstrate specific aspects of each program. This tutorial approach is novel. It is also an effective way to expose new computer users to the concepts of specific applications programs such as the three used on *Apple at Work*. After you have finished *Apple at Work*, you should have an appreciation of how much programs such as word processors, database managers, and electronic spreadsheets can accomplish for you.

Before you purchase any of these programs you should recognize and define the needs you expect the program to fulfill for you. Then you should examine the literature describing the programs that accomplish what you want. Visit a dealer and try the applications that look the closest to what you want. After you have done these things, if you still have any doubt about a particular application, find out from the dealer the name and phone number of someone who has purchased the program of interest to you. Call the people on the list and obtain their reaction to the program you are considering. If this is impractical, look for reviews of the product in computer magazines. A valuable source of information about particular products is discussions at a local Apple users group. The concept of a disk such as *Apple at Work* is excellent. It is a very concrete way to introduce particular types of applications to the new user. Chapters 9 and 10 of this book will give you further background and procedures for understanding, evaluating, and selecting software for your Apple //c.

THE INSIDE STORY

Once you have finished with the *Apple at Work* disk, you may be curious about how the Apple //c works and accomplishes all of the things it does. Take the *Apple at Work* disk out of the disk drive, place it in its protective jacket, and put it in a secure place. Take the disk entitled *The Inside Story* from its jacket and place this disk in the internal disk drive making sure that the label for *The Inside Story* is on top when you insert this disk in the disk drive. One side of this disk contains *The Inside Story*. The other side is used for *Exploring Apple Logo*.

Use the procedure described earlier (pressing open Apple, control, and reset in the order described) to perform a "cold boot" of the

Apple has provided an outstanding set of disk tutorials for the Apple //c user.

Apple //c. *The Inside Story* is an excellent demonstration of how your Apple //c functions. It uses text, graphics displays, and animation to illustrate the circuit components that make the Apple //c work. The material covered on this disk includes the loading of the operating system into the computer's memory, generation of data, the use of memory, and information flow between the computer's memory and the disk drive. This disk is another example of using a computer to teach concepts. It is professionally done and informative. If you know little or nothing about how a computer works, *The Inside Story* is an interesting and instructive look at the subject.

PROGRAMMING

Besides the disks described above that demonstrate the capabilities of the Apple //c and introduce applications software, Apple Computer, Inc. has included two disks with the system that introduce programming concepts to the new user. The first of these disks

is *Exploring Apple Logo*. The second is *Getting Down to BASIC*. Even if you know nothing about programming and do not know if you are the least bit interested in writing programs, take the time to run both of these disks. You will learn something about the principles of writing programs and something about programming languages. You may find that you are fascinated by the things you learn from your session with these disks.

Exploring Apple Logo introduces a language used by educators to teach logical thinking and the concepts of computer programming to children. This language is used in many schools in the United States. The disk is a very effective demonstration of Logo and the things it can accomplish by directing an onscreen pen carrier called a turtle to draw lines and shapes. Before you are done with the session, you will learn how to use "Turtle Graphics" to draw attractive shapes on the Apple //c's graphics screen. This disk provides an exciting way to experience something new.

The second tutorial, *Getting Down to BASIC*, illustrates the principles of Applesoft BASIC, the language permanently stored in the ROM of your Apple //c. The program on the disk uses the Apple //c to build a basic knowledge of this programming language. It presents material about BASIC onscreen, asks questions of the user, and illustrates the principles of such things as PRINT statements, computer calculation, INPUT statements, the LET statement, and several other concepts. The *Interactive User's Guide* goes a little further by providing a program listing for conversion of miles to kilometers and another program that illustrates the use of INPUT statements and the format of screen output. These programs are very short and are worthwhile practice exercises that deal with Applesoft BASIC. Chapter 8 will take you further into the exciting world of BASIC programming. This disk and *Exploring Apple Logo* are excellent devices to teach the new computer user about programs and how they are written.

SYSTEM UTILITIES DISK

When you have finished with the tutorial disks supplied with the Apple //c and have finished reading the *Interactive User's Guide*, there will be one disk left that has not been discussed. This is the Apple //c *System Utilities* disk. It is the most important tool for the Apple //c user. This disk contains utility programs for formatting blank disks, copying entire disks, renaming disks, copying files, renaming files, verifying copies, locking and unlocking files, identify-

ing and cataloging disks, and such advanced operations as configuring the serial ports of the Apple //c to use a particular device for communications or for printing. The programs on the *System Utilities* disk can be selected from a menu. The actual functions are carried out by ProDOS. The menu appears below:

```
 System Utilities                  Main Menu
   Copyright Apple Computer, Inc., 1984
-----------------------------------------

Work on Individual Files

1. Copy Files
2. Delete Files
3. Rename Files
4. Lock/Unlock Files

Work on Entire Disks

5. Duplicate a Disk
6. Format a Disk
7. Identify and Catalog a Disk
8. Advanced Operations

9. Exit System Utilities

Type a number or press the ↓  or ↑  to select an
option. Then press RETURN.
-----------------------------------------

For Help: Press    - ? or    - ?
```

This menu provides access to all the features found on the *System Utilities* disk. Help is available at any time during the use of this disk by pressing the open Apple key and the question mark together, or by pressing the closed Apple key and question mark together.

The *System Utilities* disk is so important that Apple has supplied you with a manual devoted to it. You should read the *System Utilities Manual* carefully and do the exercises. A thorough understanding of the System Utilities is the key to successful, productive, and enjoyable use of the Apple //c. We will provide a supplemental overview of the *System Utilities* below and in the next chapter. These should be read in addition to the *System Utilities Manual*, not as a replacement for it. The Copy Files option, Duplicate a Disk option, and Format a Disk option from the *System Utilities* disk may be performed using only the built-in disk drive. Although the use of a second disk drive is more convenient, it is not necessary to accomplish any of the tasks

that are done by the *System Utilities* disk. The instructions contained in The Apple //c *System Utilities Manual* are excellent. They explain the use of every function performed by the programs on the *System Utilities* disk. Rather than discuss every single step in great detail, we will discuss a few of the more important functions performed by programs on this disk.

Most of the functions on the menu are covered in greater detail in the next chapter including copying procedures, formatting disks, locking and unlocking files, and so on. One of the menu options on the *System Utilities* disk is unique. The option called Identify and Catalog a Disk can examine a disk that is formatted in DOS 3.2, DOS 3.3, or ProDOS. It can display a listing of the programs on the disk being examined. The Advanced Operations option allows the user to set ProDOS prefixes, create ProDOS subdirectories, verify that a disk is readable, change a disk's format, and configure the computer's serial ports. These options are vital to the ProDOS user and to the serious printer and modem user. They provide a convenient method of realizing the full power of your Apple //c.

The process of configuring or setting up your computer to deal with serial devices such as printers and modems is handled very well by the *System Utilities* disk. You are asked a series of questions by the program. Each question defines a specific requirement of the device you will be connecting to serial port 1 or serial port 2. The correct answer to each of these questions is found in the manual supplied with the serial device you are setting up the Apple //c to handle. If you follow the onscreen instructions presented by this program and consult the device manual for a serial device, you will have no difficulty making sure that the Apple //c uses the device properly. The most common configuration is pre-set so you may never need to change anything.

CONCLUSION

The *Interactive User's Guide* and the disks that accompany it allow the new Apple //c system user to learn how to operate the system more effectively and properly. They also expose the new user to applications programs and to programming concepts and languages. This method of demonstrating the potential of the Apple //c is unique and exciting. Apple is the only computer company to reach this level of expertise and sophistication in providing excellent disk-based tutorials to the user. It is certainly a valuable part of the entire Apple //c system. The *System Utilities* disk provides the powerful functions of ProDOS, the Apple //c's modern Disk Operating Sys-

tem, in the form of an easy-to-use menu. These outstanding easy-to-use disks illustrate Apple Computer, Inc.'s commitment to making the time you spend with a computer a time that you enjoy rather than a time of frustration.

5

The *System Utilities* Disk

FILE MANAGEMENT ON THE APPLE //C

Now that you have been exposed to the capabilities of the Apple //c through the tutorial disks, you probably want to get to work with your new machine. If you have purchased any commercial software such as Applewriter, Appleworks, Datafax, MultiPlan (DOS 3.3 version), or other software, you will need a few blank disks to store files from these programs. Before a blank disk can be used to store files, it must be formatted to accept files from the same disk operating system used by the program that writes the files. Fortunately, many applications programs permit you to format blank disks directly from the applications program so that you never need to know what format the disk should be.

Formatting is the process of magnetically or electronically marking a disk to allow a disk operating system to read and write data to the disk. Apple Computer, Inc., has thoughtfully provided a disk to do such things as format disks, copy files from one disk to another, rename files, verify disks, and configure the serial ports of the Apple //c. This disk is called the Apple //c *System Utilities* disk. This disk contains the ProDOS Disk Operating System and has the tools you will need to accomplish the tasks already mentioned here and a few more tasks that were not mentioned.

The *System Utilities* disk is extraordinarily easy to use because the programs on the disk lead you through each step of each process in an extremely clear fashion. They even have Help screens available if you press the open Apple key or the closed Apple key and a question mark at the same time. This disk is really fun for the experienced user who has had some exposure to DOS 3.3 and the *ProDOS User's Disk*. The Apple //c *System Utilities* disk is a collection of programs that allows you to manage the information being stored on your disks. As you become more adept at the use of your system, you will use this disk frequently to manage and copy the files that you have generated as you use applications programs and write your own programs. Now let's get down to the fun of using this extremely valuable and powerful tool.

PUTTING THE *SYSTEM UTILITIES* DISK TO WORK

Take the *System Utilities* disk from its jacket and insert it into the internal disk drive of the Apple //c carefully. The label should be up and the read/write slot should be facing the computer. Close the disk

drive door. Turn on the computer's power switch. You will see the following screen display, after a brief message, that displays the version of ProDOS that is used as a disk operating system by the programs on the disk:

```
System Utilities                    Main Menu
  Copyright Apple Computer, Inc., 1984
----------------------------------------


Work on Individual Files

1. <COPY FILES>
2. Delete Files
3. Rename Files
4. Lock/Unlock Files

Work on Entire Disks

5. Duplicate a Disk
6. Format a Disk
7. Identify and Catalog a Disk
8. Advanced Operations

9. Exit the System Utilities

Type a number or press ↑  or ↓  to select an
option. Then press RETURN. -
----------------------------------------

For Help: Press   -? or   -?
```

The screen and those that follow look slightly different if you are using a 40-column display. You should get into the habit of backing up those disks that can be backed up with the *System Utilities* disk, especially disks that contain important material such as the *System Utilities* disk. If there are several members of your family who have not used the tutorial disks yet or if you expect to refer to these disks frequently, you should back up these disks as well. If you wish to back up both sides of the double-sided disks, you would use a backup disk for each side. Your first effort should be the *System Utilities* disk, one of the most important disks in your collection.

The first step in copying a disk must always be to place a write-protect tab over the notch in the disk. If there are two notches, tape over both of them since you could inadvertently put the disk in upside down during the copy operation. This step is essential because during the copy process you will have to swap disks several times. The possibility of putting the wrong disk in at the wrong time,

and thereby destroying the data you were trying to copy, is too great to *ever* copy a disk without write-protecting it first by placing a write-protect tab on your valuable original disk. The computer will detect any attempt to write on the original disk and prevent loss of your data. **Never copy a disk without protecting it with a write-protect tab first.** Equal care is required to ensure that you do not inadvertently copy *onto* a disk that has valid data. This requires that you carefully check the contents of the destination disk (the one you will copy to) to verify it is not needed (or that you are using a new diskette). More on this later.

Select option 5 from the Main Menu. You can do this by pressing the 5 or by pressing the down arrow key until you have reached the line where the highlight is on the description DUPLICATE A DISK. When the highlight is on this line, it will appear in all capital letters as opposed to upper and lower case letters on the rest of the lines. It will also be displayed between the left and right angle brackets (< >). After you press RETURN, the following screen will be displayed:

```
System Utilities            Duplicate a Disk
Version 1.0                   ESC: Main Menu

----------------------------------------

Where is your source disk?

1. <BUILT - IN DRIVE>

2. External Drive




Type a number or press ↓ or ↑ to select an option.
Then press RETURN. -

----------------------------------------

For Help: Press  -? or  -?
```

If for some reason you are not sure of what you are doing you can use the Help feature of the *System Utilities* disk by pressing the open Apple key and the question mark at the same time. This will cause a window (rectangular box) to be displayed in the upper right hand

section of the display illustrated above. A message in this window says:

```
"The source disk is the disk you
want to duplicate. Choose option 1
if you intend to put the source
disk in the built-in disk drive.
If you have an external disk
drive, you can choose option 2.

Press RETURN to continue."
```

This Help screen is an extremely friendly feature of the *System Utilities* disk. Press the RETURN key to get back to the selection process.

Press the RETURN key to specify that the disk you wish to copy is in the built-in disk drive. This will cause another display, which looks like the one shown below, to be on the screen:

```
System Utilities                      Duplicate a Disk
Version 1.0                             ESC: Main Menu

------------------------------------------------------

                  Source Disk:Built-in Drive

                  Where is your destination disk?

                  1. <BUILT-IN DRIVE>

                  2. External Drive



Type a number or ↓ or ↑ to select
an option. Then press RETURN

------------------------------------------------------

For Help: Press  -? or  -?
```

Press RETURN while the highlight is resting on the BUILT-IN DRIVE option. This will cause the screen shown below to appear:

```
    System Utilities
    Duplicate a Disk
Version 1.0                                   ESC: Main Menu

 --------------------------------------------------------

                 Source Disk : Built-in Drive

                 Destination Disk : Built-in Drive


 ------------------------------------------------------
 :                                                    :
 : Place the source disk in the Built-in Drive.       :
 :                                                    :
 :                                                    :
 :                                                    :
 :                                                    :
 : Press RETURN to continue;                          :
 : ESC to return to the Main Menu.                    :
 :                                                    :
 ------------------------------------------------------
```

At this point, you should press the RETURN key. The next screen will ask you to place the "Destination Disk" in the internal drive and press RETURN again causing the screen below to be displayed with the default name "/UTILITIES" supplied for the new disk:

```
System Utilities                          Duplicate a Disk
Version 1.0                                 ESC: Main Menu

 --------------------------------------------------------

         Source Disk : Built-in Drive

         Destination Disk : Built-in Drive

         Operating System : ProDOS

         Enter Name of New Volume: /UTILITIES.....




 --------------------------------------------------------

For Help: Press  ⌘-? or  -?
```

Since you are making a copy of the *System Utilities* disk, press RETURN at this point. If the destination disk is not blank, a cautionary message is printed on the screen. In effect, the message says that there is an already formatted disk in the disk drive and asks you if you wish to destroy the files on the disk. The highlight is on the "Yes" response. If you press the RETURN key with such a disk in the disk drive, the contents of the disk will be erased and replaced with the data from the source disk you are copying. If you move the pointer to the "No" response, the copy sequence will be aborted and you will be returned to the screen that requests the location of the source disk. You will be required to restart the sequence at that point.

Since the disk you are using for the copy should be blank, this sequence of events should not arise. There is a remote possibility you may have acquired pre-formatted disks from a dealer or friend. In this case, *System Utilities* will ask "Is it OK to destroy" followed by the name or type of disk. If you answer yes, the format steps will proceed as usual. If you are quite sure there is no data on the disk, proceed with the copy. If you have even the slightest doubt, press ESCAPE to return to the main menu. You can use the "Identity and Catalog a Disk" option (#7) described below to see a list of the file names on the disk. This may help you decide if you really want to erase the disk by copying another disk onto it.

It may be useful to think of disks as being similar to tape recorder cassettes. If you have an album recorded on a cassette, you can record another album right over it, erasing the original recording. This can happen intentionally, because you are tired of the original recording, or accidentally because you did not realize that the cassette had music on it. When you copy one entire disk onto another, the original contents of the destination disk are destroyed, being replaced with a copy of the source disk. Formatting also erases the output disk. These are points where there are risks of losing valuable programs and data, so be *extra*cautious. Later, when you have amassed more disks than you have now, it is quite possible you may confuse two disks and need to take this "bail out" opportunity.

For now, press the RETURN key and the copy process will start. The next message you see on the screen will be a message that says "Formatting." This process is carried out automatically whenever an entire disk is copied to a disk by the Apple //c's *System Utilities*. When the destination disk has been successfully formatted, a "Done" message is displayed.

The copy program uses a portion of RAM to hold the data to be copied. When the dedicated portion of RAM is full, the program instructs you to remove the source disk and to replace it with the destination disk. The contents of memory are then written to the des-

tination disk. As soon as all of the data has been copied onto the destination disk, the program instructs you to replace the copy disk with the source disk. This cycle is repeated until the contents of the source disk have been copied to the destination disk. Once you have completed this operation you will appreciate the absolute necessity of putting the write-protect tab on the notch of the original, valuable disk. As soon as the copy has been made, remove the original *System Utilities* disk from the internal disk drive and store it in a safe place. Use the copy for future file handling activity. Label the new *System Utilities* disk using the precautions described earlier (do not write on the disk itself with anything except a soft-point, felt-tip pen. The new *System Utilities* disk should, from this point, be used in the internal drive.

If you have two disk drives, the Duplicate a Disk operation is much more simple and faster. The entire operation takes place at one time without disk swapping. The only difference in procedure is that you specify one of the disks as being in the "EXTERNAL DRIVE."

The *System Utilities* disk can use the "Duplicate a Disk" option to copy disks that have been formatted on DOS 3.3, ProDOS, and Pascal. The operating system used by such disks will be available on the copy as long as it was originally available on the source disk.

FORMATTING DISKS

In the first paragraph of this chapter, we discussed the need to format disks to allow the files and data generated by various programs to be preserved. In most cases, the disk used to store data must use the same operating system as the program that produces and saves the data. The *System Utilities* disk is capable of formatting disks in the format required by DOS 3.3, Pascal, and ProDOS. This important and convenient formatting capability eliminates the need for a DOS 3.3 System Master disk. Formatting a disk is the process of getting a blank disk ready to accept data from a particular disk operating system such as DOS 3.3, ProDOS, or Pascal. This option is exercised by selecting option number 6 from the *System Utilities* disk.

Formatting is accomplished by electronically marking the sector locations on the surface of the disk and setting up a means of locating files on a disk. This could be a ProDOS volume directory, which is actually a file, or a DOS 3.3 directory that is not a file. The screen displays that are associated with formatting disks are similar to the screen displays that were used to illustrate the operation of the Copy a Disk option. This process may be carried out with an internal or external disk drive. The program tells you when to swap the original

disk for the disk that is being formatted. This feature of the *System Utilities* disk is extremely convenient. Once again, be extra careful that the disk you format does not contain any valuable data or programs, as it will be erased.

IDENTIFY AND CATALOG A DISK

The "Identify and Catalog a Disk" option is used to display a list of the files on a disk. To test this option, select option "7" from the *System Utilities* disk Main Menu. This selection will cause the following screen to be displayed:

```
System Utilities              Identify and Catalog a Disk
Version 1.0                                ESC: Main Menu

-------------------------------------------------------

                    Where is your disk?

                    1.   <BUILT-IN DRIVE>

                    2.   EXTERNAL DRIVE

                    3.   ProDOS pathname




Type a number or press ↓ or ↑ to select
an option. Then press RETURN.

-------------------------------------------------------

For Help Press   -? or   -?
```

To test this option, remove the *System Utilities* disk from the internal disk drive of the Apple //c and replace it with the "Apple at Play" disk. Accept the selection of the internal disk drive by pressing RETURN. The following display will appear on the screen:

```
System Utilities          Identify and Catalog a Disk
Version 1.0                           ESC: Main Menu
----------------------------------------------------

Disk Name : /APPLE.PLAY        Disk Format : ProDOS
Filename               Type       Size
*PRODOS               ProDOS       29
*BASIC.SYSTEM         ProDOS       21
 PLAY.INTRO           Text          4
*CM.OBJ               Binary        3
 LEMONADE             ABasic       22
 FUN.W.FINANCE        ABasic       24
 PLAY                 ABasic        3
 PLAYER2              Binary        1
*SONATA               Binary       10
*CARDCHARS            Binary        8
*PU                   Binary        3
*APPLE.21             ABasic       15
*SMD.JMP              Binary        1
*SMD.OBJ              Binary        3
*QUIT                 ABasic        1
*STARTUP              ABasic        6
*QUIZ                 ABasic        9
 SPACE.QUARKS         ABasic        1
 QUARK LOGO           Binary        5
 PRACTICE.TEXT        Text          3
 SQ1                  Binary       28
 PLAY.MUSIC           ABasic        1
 SD2                  Binary       44
 UP                   Binary        1
 SP.DOWN              ABasic        1
 Q.CH1                Text          7
 Q.CH2                Text          5
 Q.CH3                Text          6
 Q.CH4                Text          5

29 Files Listed, 270 Blocks Listed, 2 Available

----------------------------------------------------

Listing complete; Press RETURN to continue; ESC to
return to the Main Menu
```

This catalog of the "Apple at Play" disk illustrates a number of characteristics of the catalogs of disks generated by the *System Utilities* disk. First, the name (if any) of the Volume (disk) is displayed in the upper left hand corner of the catalog. The disk format is then displayed in the upper right hand corner of the catalog. Filenames

are listed in the left hand column. Those files that are marked with asterisks are locked, which means that these files may not be deleted until they have been unlocked. Locked files may not be renamed. Their contents may not be changed. A ProDOS filename must begin with a letter. It may also contain numbers and periods. There may be no spaces or punctuation marks other than periods in a filename. ProDOS filenames may only be 15 characters long. Pascal filenames may only be seven characters long.

DOS 3.3 filenames may be 30 characters long and may contain spaces. However, in the use of the Apple //c and its *System Utilities* disk, you would have less chance of encountering problems with filenames if you observed the more restrictive rules for ProDOS filenames even on DOS 3.3 disks. The center column lists the file type for each file on the disk. The list includes commonly encountered file type abbreviations. The right hand column lists the size of each file in blocks. A block is made up of 512 bytes of data. Each character in a file takes up a byte of data. The message at the bottom of the catalog recapitulates the number of files listed, the number of blocks listed, and the number of blocks available. For a DOS 3.3 file, the volume name would be the volume number assigned by the user or the default value of 254. A default value is a value assigned by the programmer as a best guess as to what your response will be. It could also be a safe value that will not cause problems with the system.

COMMON FILE TYPE ABBREVIATIONS—*SYSTEM UTILITIES* DISK

Abbreviation	Description
ABasic	Applesoft BASIC program
Text	Text file
Binary	A machine language program, a binary image of a file
ABasicV	Applesoft BASIC Variables
Dir	A ProDOS Directory
ProDOS	A ProDOS system file
User=F1	User defined code
RelCode	A relocatable code file

These abbreviations are examples of what you might encounter. Note that these abbreviations are specific to the *System Utilities* disk. The *ProDOS User's Disk* and DOS 3.3 have different abbreviations

for the same file types—see Chapter 6. We found these file types on the various disks supplied with the Apple //c system.

COPY FILES

The "Copy Files" option from the *System Utilities* disk is an option you will probably use often. This option allows you to copy individual files to back up important data and programs. If you discover a few bad blocks on a disk when you verify that a disk is readable, use this option to copy those files that are still good to another disk. You might want to share a program that you wrote with a friend. You may have written an article with a word processor and wish to submit it to a magazine on disk. The "Copy Files" option gives you the ability to keep the original file and make copies of the original for the reasons discussed here. One thing we cannot stress enough—back up important documents. If you have generated a profit and loss statement with a spreadsheet program, this document has some value to you. Back it up by making a copy and then store the disk containing the copy in a safe place. The same principle applies to the mailing list or telephone list you generated on your favorite database management program.

This option provides screens that lead you through each step of the process. Read the screens carefully and follow the instructions pertaining to your situation. If necessary, at any point during the process, consult the "Help" feature by pressing the open Apple key or closed Apple key and the question mark. The "Copy Files" option and the rest of the file handling options present the user with a very convenient method for selecting files for file handling operations.

After you have selected the "Copy Files" option, a file selection screen is displayed. This screen lists all of the files on the source disk and allows you to select the files you wish to copy, rename, delete, unlock, or lock. A file is selected by scrolling down the list until the selected file is highlighted. Then the right arrow key is pressed to select files. Selected files are marked with a check mark. If you change your mind and decide that you have marked the wrong file, the left arrow key is used to unmark a file. This approach is really helpful. There is nothing more agonizing than telling the computer to copy a file from one disk to another, then discovering that you have forgotten the exact name of the file. If you mark more than one file, the program copies all of the marked files rather than making you select a file at a time. This approach to file handling is exceptional.

DELETE FILES

The "Delete Files" option allows you to remove files from a disk directory. This option makes space on the disk by allowing you to delete those files you no longer need. You might also use this option to delete those files that have bad sectors when you use the "Verify That a Disk is Readable" option. As with the other options on the disk, this option uses very clear screens to present various courses of action to you. "Help" screens are also available if you have any doubt about what to do next. The same file selection procedure is used with this option as is used with the "Copy Files" option.

RENAME FILES

The "Rename files" option allows you to change the name of any unlocked file on a disk. You may have written a program you call "MortgageCalc." Later you think of a better way to do the job, so you write an improved version. At the same time, you want to preserve the earlier version because it has a few features you did not put in the second version. You also want to call the second version "MortgageCalc" because it is easier to remember. However, if you save the new file with the same name as the old file, it will overwrite the old file. You should give the original file a new name such as "Mortcalc." You would use the "Rename Files" option to make the change in name.The program screens lead you through each step of the process. If you read these screens carefully and provide all of the required information, the process will operate smoothly.

LOCK / UNLOCK FILES

The "Lock/Unlock" option on the Main Menu allows you to either lock or unlock the files on a disk. The process of locking files helps to ensure that important data cannot be erased from the disk without first unlocking it (except in the formatting or duplicating a disk process). The "Lock/Unlock" option protects your files and should be used in situations where you are afraid that a file will be renamed, written to, or erased. Locked files may not be deleted (erased) or renamed.

ADVANCED OPERATIONS

The only option that we have not discussed so far besides the option to "Exit System Utilities" is the "Advanced Operations" option. This option is used by pressing the "8" key when the Main Menu is being displayed. This selection process will cause another menu to be displayed on the screen. This menu is presented below:

```
System Utilities                    Advanced Operations
Version 1.0                              ESC: Main Menu

--------------------------------------------------------

                ProDOS Only

                1.  <SET THE PREFIX>
                2.  Create a Subdirectory

                Additional Operations

                3.  Change a Disk Format
                4.  Verify That a Disk is Readable
                5.  Configure the Serial Ports


Type a number or press ↓ or ↑ to select
an option. Then press RETURN.

--------------------------------------------------------

For Help : Press   -? or   -?
```

ProDOS OPTIONS

In the Advanced Operations menu just discussed the first two options are used only with ProDOS disks. ProDOS requires that each volume or disk be given a name. It also requires that instructions associated with file handling contain not only the name of the file, but also the name of the volume. (Two exceptions are: 1) if the volume has been included in a previous SET THE PREFIX operation and 2) you can use the slot and drive format.) A volume is assigned a name when a disk is formatted for the ProDOS disk operating system. Files may also be arranged into subdirectories on ProDOS disks. A subdirectory may be created to associate files of a like type or files related to a particular

operation. If files are placed in subdirectories, ProDOS requires inclusion of the subdirectory name in any instructions dealing with a file. The order of events in this system is:

Volume Directory/Subdirectory/Filename

There is no absolute need for you to use subdirectories and pathnames. If you are new to computers do not use them until you feel you are ready. The menus provided by the *System Utilities* provide all of the functions you will normally need without using subdirectories or pathnames. If you use subdirectories, then you will need to use pathnames to refer to files in subdirectories. The Volume Directory/Subdirectory/Filename line above illustrates the concept of pathnames. The full name required to access a file is called a pathname. If you use subdirectories, the pathname could contain all of the elements in the illustration.

ProDOS provides an abbreviated way of specifying pathnames. This feature is useful because pathnames can require quite a few keystrokes; and because the Volume Directory name and/or Subdirectory name may stay the same for several subsequent operations. ProDOS allows you to establish a "Prefix" that will be added in front of any pathname you use. Suppose, for example, that you have organized several files on your copy of the *System Utilities* disk into a subdirectory called "HOLD." The full pathname for a file named TEST when located in subdirectory HOLD on volume UTILITIES is: /UTILITIES/HOLD/TEST. To save time typing the first part (prefix) of this pathname, use the "Set Prefix" function to establish "/UTILITIES/HOLD" as the prefix. You can then enter TEST as a pathname. ProDOS will automatically add the current prefix (/UTILITIES/HOLD) in front of what you type to establish the complete pathname: /UTILITIES/HOLD/TEST.

Prefixes are an implied or "automatic" part of a pathname. They can be set to save time for the user. When a prefix is set, it remains set until it is changed, regardless of the file handling operations that are carried out. The prefix in the above illustration could consist of just the volume name or of the volume name and the subdirectory name. The "Set the Prefix" option of the Advanced Operations Menu allows the user to set prefixes for his or her convenience while copying, renaming, locking, or unlocking files stored on ProDOS formatted disks is taking place. The "Create a Subdirectory" option allows the user to create a directory file on a ProDOS formatted disk. Then as files are saved or copied, they may be associated with this subdirectory. ProDOS will be dealt with in greater detail later in this book. For beginning users, we recommend that you stay away from

subdirectories and pathnames until you are thoroughly familiar with the *System Utilities* disk, ProDOS, and general computer operations.

ADDITIONAL OPERATIONS

Two of the additional operations on the Advanced Operations menu deal with disks. The third deals with setting up the serial ports used to communicate with peripherals such as printers and modems. The first of the disk handling options is "Change a Disk's Format." If you select this option, the following screen is displayed:

```
System Utilities                  Change a Disk's Format
Version 1.0                      ESC:Advanced Operations

--------------------------------------------------------

            Which conversion do you want to make?

            1.  <DOS 3.2 -> DOS 3.3>

            2.   DOS 3.3 -> ProDOS

            3.   ProDOS -> DOS 3.3




Type a number or press ↓ or ↑ to select
an option.

--------------------------------------------------------
```

The Change a Disk's Format option lets you convert disks that have been formatted under one disk operating system to the format of another disk operating system. Conversions from DOS 3.2 to DOS 3.3 can be made quite safely. They would only be made if you encountered an old disk from someone's collection of early programs for the Apple II or Apple II+ computers. You will probably use the DOS 3.3 to ProDOS conversion and the ProDOS to DOS 3.3 conversion more frequently than the first option. This conversion process may be carried out on the Apple //c with only the built-in disk drive, or by using the internal disk drive for one of the disks and your external disk drive for the other.

The program that carries out the conversion process provides you with clear instructions to ensure that you carry out the process suc-

cessfully. This particular program is another indication of Apple's commitment to compatibility in its product line. The company has ensured that programs and data files may be transported from earlier versions of the operating systems used on its computers to later versions wherever possible. As new operating systems are introduced, tools are provided to link them with past operating systems within a family of computers.

VERIFY THAT A DISK IS READABLE

The "Verify that a Disk is Readable" option is a useful tool on the Advanced Operations Submenu. This option can be used to check newly formatted disks to ensure that they are not damaged in any physical or magnetic way. It may also be used to examine disks that are in use. If you discover a number of bad blocks on a disk you are using for files, examine each file and run each program on the disk to identify those that occupy bad blocks. Use the "Copy Files" option from the main menu of the *System Utilities* disk to copy good files and programs to a good disk. After this step, reformat the damaged disk and verify it again. If it is still bad, throw it away. You may be able to view at least some parts of files or programs on the damaged disk using a disk dump utility such as the one provided on the *Apple //c Explorer's Disk* sold in conjunction with this book.

CONFIGURE THE SERIAL PORTS

The final option on the Advanced Operations Menu is the "Configure the Serial Ports" option. This option allows you to ensure that the Apple //c compatible peripherals you use will communicate correctly with the Apple //c. The option is unnecessary for the use of Apple products such as the Imagewriter and the Apple 300 Modem. If, however, you decide to purchase a peripheral made by another manufacturer, it may have slightly different rules of communication than the Apple products. This option allows you to adjust the rules of communication or protocol used between the Apple //c and the peripheral. You should refer to the *System Utilities Manual* for instructions for changing configuration. You should also refer to the instruction manual for your particular machine for the detailed information required to configure the system properly. The "Configure the Serial Ports" option is extremely easy to use with the *System Utilities Manual* in hand and the device manual available for reference purposes. This option uses the same structured approach as the

other options described in this chapter. Configuration is a one-time process when you connect a new peripheral to the system. Once the configuration has been saved on the *System Utilities* disk, it is loaded into the computer's memory each time the disk is booted. You can seek additional help from the dealer who sold you the peripheral or from an experienced user.

SUMMARY OF THE *SYSTEM UTILITIES*

As you can see from this discussion, you have an extremely valuable tool in your *System Utilities* disk. File handling operations, disk handling operations, operating system conversion, and system configuration are supported by programs on this disk. All operations are carried out in a structured fashion assisting the user through each step of each process. The *System Utilities* disk uses the ProDOS operating system to carry out its functions as they relate to disk drives. The functions carried out by the programs on the disk do not include the loading, running, and saving of individual programs. These operations are carried out by using the ProDOS commands that are available from Applesoft BASIC. The principles that underly the ProDOS operating system have not been covered in this chapter. If you wish to have a better grasp of how this exciting operating system works, see Chapter 6 which compares ProDOS to DOS 3.3. The following pages deal with the *ProDOS User's Disk*. This is an optional, extra-cost alternative to the *System Utilities*. If you are a beginner at computing, you will probably want to save these optional considerations for later and go on to the last section in this chapter, "Where to Go From Here?"

ProDOS ON THE *SYSTEM UTILITIES* DISK VS THE *ProDOS USER'S DISK*

ProDOS is currently available to the Apple //c user on two main utility disks. The *System Utilities* disk is supplied with the Apple //c. The *ProDOS User's Disk* is also available at additional charge. Each of these disks uses a different approach to making the features of the operating system available to the user. The underlying system on both disks is the same, but the way that it is presented to the user via a system of menus and submenus is different. Certain options available on the *System Utilities* disk are not available on the *ProDOS User's Disk*, and vice versa.

If you are a complete beginner with computers, you may want to only read about the *System Utilities* for now. The following pages will contrast the *System Utilities* with the optional *ProDOS User's Disk*. Feel free to skip over them for now and return to them later when you have more computer experience under your belt. The same considerations apply to Chapters 6, 7, and 8. These cover topics that are not crucial to your first weeks of computing activites. If you are looking for depth of understanding, they will broaden and deepen your understanding and knowledge. There is much to be said, however, for taking a break at this time to enjoy your new Apple //c computer and allow all of this new information and terminology to sink in. Chapter 8 deals with writing your own programs in Applesoft BASIC, which you may or may not want to start at this time.

The *System Utilities* disk is easier for the inexperienced user to learn than the *ProDOS User's Disk*. Its menu program is structured to provide choices and guidance in easy-to-understand terms. The words used to name the various options, to explain them, and to request specific courses of action from the user are not as technical as those on the *ProDOS User's Disk*. This comment is not negative regarding the *ProDOS User's Disk*, because the menu presentation on this disk is excellent. It is slightly more condensed than the presentation on the *System Utilities* disk. As you become more experienced in the use of your Apple //c, you may wish to purchase the *ProDOS User's Disk* and the manuals accompanying it.

FILE TYPES

The *ProDOS User's Disk* uses three-character symbols to identify 256 different file types for the user The *System Utilities* disk displays a different set of abbreviations in the catalog that is displayed from the "Identify and Catalog a Disk" option on the Main Menu. These abbreviations were listed above. Some of the *ProDOS User's Disk* file type abbreviations are user definable. The list below demonstrates the most common file types displayed by the *ProDOS User's Disk*.

Abbreviation	**Meaning**
TXT	ASCII Text file
BIN	Binary file
BAS	Applesoft BASIC Program
VAR	Applesoft Variables
REL	Relocatable Code
SYS	ProDOS System File

These file types are also used in CAT and CATALOG displays from Applesoft with ProDOS and BASIC. SYSTEM regardless of which utility disk you are using.

MEMORY USE

ProDOS configures part of the Apple //c's expanded memory as if it was another data storage device called RAM. This means that programs and data can be transferred from disk to this section of memory. They can then be accessed by those programs that are running as if they were accessing another disk drive. If nothing else, this process will increase the speed of file device access for those files or programs that are transferred to this section of memory. We have used this feature to assist with single-drive file conversion and to speed data access on the Apple //c computer. The volume name given to this portion of memory when it is configured in this fashion is /RAM.

The convention for naming this section of memory as a storage device is to address it as slot 3, drive 2 or as /RAM.

The *System Utilities* disk uses the extra RAM in its own operations. The use of the extra RAM space as a storage device is not available from the *System Utilities* disk menus. The /RAM feature is available in Applesoft when you exit from the *System Utilities,* just as with the *ProDos User's Disk.*

The *System Utilities* disk provides the user with Help screens at all points along the way. The *ProDOS User's Disk* provides tutorials at the start of each menu to provide the user with guidance.

ProDOS is exactly the same operating system whether you use it with the *System Utilities,* with the *ProDOS User's Disk,* through the Applesoft BASIC interface provided by BASIC.SYSTEM, or through application programs you have purchased or written yourself. Since most beginners will use one of the two major utility disks to perform routine disk operations, we have tried to give you some background for understanding the two options provided by the *System Utilities* provided with the Apple //c and the *ProDOS User's Disk,* which you can purchase separately. The only significant reason you might want to purchase the *ProDOS User's Disk* would be to use its /RAM capability.

WHERE TO GO FROM HERE?

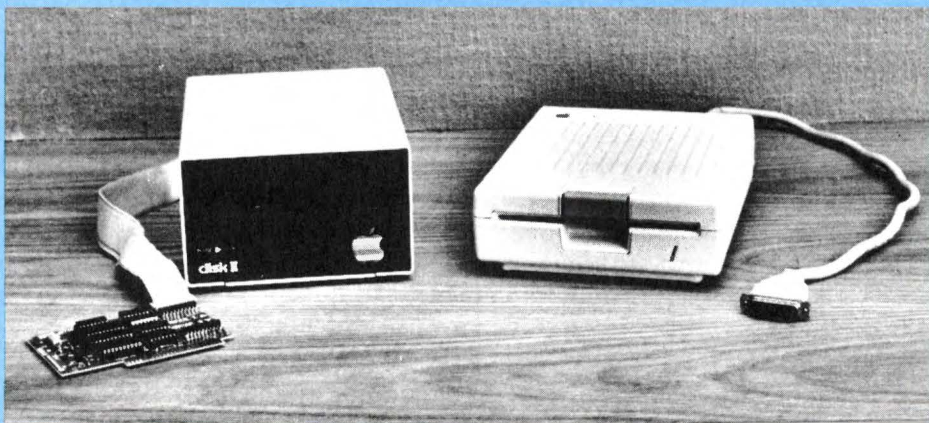
For your first learning period with the Apple //c, it is probably best to stick with the *System Utilities.* Later, as you grow in experi-

ence and knowledge, you may return to this chapter and Chapter 7, which deals with the *ProDOS User's Disk* in more detail. Perhaps you may eventually choose to use the *ProDOS User's Disk* as a supplement to the fine *System Utilities* provided with the Apple //c.

The following chapter also deals with a topic that is not required for beginning use of the Apple //c: the differences and similarities between ProDOS and the older Apple DOS 3.3. If you are already an Apple user with DOS 3.3, or plan to use a number of DOS 3.3 files or programs, this material will be of special interest. Otherwise, it is probably better to leave it for a future time when you are comfortable with the fundamentals of the Apple //c and want to stretch the horizons of your knowledge and understanding. At this point you may want to proceed to Chapter 7 to learn more about the *ProDOS User's Disk*, skip to Chapter 8 to learn more about programming in Applesoft BASIC, or skip to the latter part of this book to learn more about the software and hardware available for the Apple //c.

6

ProDOS vs DOS 3.3



DISK OPERATING SYSTEMS

The Apple //c computer can use a number of disk operating systems that have been introduced by Apple Computer, Inc. This versatile computer uses the current operating system called ProDOS. It can also use ProDOS's immediate predecessor, DOS 3.3, and the Pascal operating system. ProDOS is the newest and the most powerful of the three operating systems. DOS 3.3 has been in use since 1980 and is used by thousands of programs that are available on the market for the Apple II family of computers. This family includes the Apple II, the Apple II+, the Apple //e, and the Apple //c. Of the three operating systems available to you as an Apple //c user, ProDOS should be your first choice if you are a new user and if you plan to grow with the system you are using. We stress this fact because we have been involved with personal computers for a number of years and have had the opportunity to observe the development and use of software to meet the needs of changing systems. Software authors and manufacturers always try to keep up with the most popular hardware and powerful systems that are available to the consumer. In addition, ProDOS is supplied with the Apple //c on the *System Utilities* disk. Other operating systems would have to be purchased separately.

ProDOS is the standard disk operating system of the Apple //c. It is supplied with the computer on the *System Utilities* disk. Most applications programs incorporate the disk operating system or have the ability to access the command structure of the operating system used by a computer. Most applications programs save the data that they generate only to disks that have been formatted to use the same operating system that the applications program uses. This means that a particular word processor, database manager, or spreadsheet program could generate files in a DOS 3.3 format, a ProDOS format, or a Pascal format, depending upon the operating system employed by the program.

Many programs do not allow you to format disks from within the program. These programs require you to supply a disk for storing files that is properly formatted to use the operating system used by the program. If you did not have the *System Utilities* disk, you would probably need to purchase a copy of DOS 3.3 to format DOS 3.3 data disks. The term "formatting," or its twin "initializing," is used to describe the process of preparing a disk to accept the files and programs that are used by a computer's operating system. These terms will be described in greater detail in the next chapter of this book. In the meantime, let's examine ProDOS and DOS 3.3 by comparing these different operating systems.

DISK FORMAT AND USE

Both ProDOS and DOS 3.3 use 35 tracks on a 5¼-inch floppy disk. Both operating systems use 16 sectors per track for storing data and files. Each of these sectors can hold 256 bytes of data. DOS 3.3 storage units are called sectors and utilize a single sector as a storage unit. ProDOS storage units are called blocks. These blocks are made up of two sectors and use 512 bytes of space. Both operating systems number each track from the outside of the disk toward the inside starting at the outside with track "0" and finishing at the inside with track "34." Each track and sector is marked by the disk drive's read/write head when the disk is formatted by the operating system.

If you multiply the number of bytes times the sectors times the tracks, you will find that both operating systems provide the user with a maximum storage capacity of 143,360 bytes of data per disk. Not all of this capacity is available as some space is used by the disk operating system. Some space is used by files called volume directories in ProDOS. In DOS 3.3 there are no directory files. Instead, a specific part of the disk is dedicated to storing key pieces of information about files. This information is used by DOS to locate files on a disk and to use them.

THE USER INTERFACE

The user interface of a program is the method used by the program to get information and commands from the program's user. DOS 3.3 is a command-driven operating system. This means that the user gives the system its instructions by typing a command in a specified way followed by pressing the RETURN key. The computer acts upon the command to perform the function selected by the user. ProDOS is accessed via machine language interface calls. A command interface for BASIC, called BASIC.SYSTEM, is also available. This makes ProDOS look like a command-driven disk operating system and very much like DOS 3.3. It is command-driven in the sense that it will respond to commands entered by the user through Applesoft and BASIC.SYSTEM.

DOS 3.3 is actually an assembly language program. An assembly language program is a program that uses special mnemonics to give the computer instructions. It is one step above the machine language that the computer understands and a step below the command words that most users understand. As with most symbolic languages, assembly programs are translated into machine

language. You load this program into the computer's memory by booting the DOS 3.3 *System Master* disk. It occupies about 10K of Random Access Memory space in the computer's memory. It prevents the computer from allowing data to be written in the memory space that it occupies by setting aside the highest location of memory available for data, files, and programs.

This assembly language program must be available in memory for DOS 3.3 to work. There is a program on the DOS 3.3 *System Master* disk called FID, which provides an interface between the user and the operating system for copying and manipulating files. Two other programs are available to provide the necessary interface involved in copying entire disks. These programs are called COPYA and COPY.

ProDOS is loaded into the Apple //c's memory from a ProDOS startup disk or from an applications disk that contains the program called ProDOS. A second program called XXX.SYSTEM is required on the ProDOS startup disk. XXX.SYSTEM is a system program that provides the interface between the system user and the computer. The system that you will encounter most frequently as a new user is a BASIC.SYSTEM user interface.

DIRECTORIES AND CATALOGS

DOS 3.3 uses a single CATALOG command to list the files on a disk. This listing is a sequential list of files that identifies the write-protect status of the file, the file type, the filename, and the file length in sectors. This catalog is also available from FID, the file copy program.

ProDOS will display two types of catalogs from BASIC. One of these catalogs is a 40-column catalog that displays the Volume Name of the disk, each filename, the file type, the file length in blocks, the date the file was last modified, the blocks of space on the disk that are used, and the blocks of space that are free. The 80-column catalog also provides the starting address of the file, the date the file was created, and additional storage information.

A DOS 3.3 catalog can contain up to 105 filenames. The ProDOS Volume Directory can only contain 51 filenames. All or some of these filenames may be subdirectories that are files containing lists of other files associated in a fashion selected by the user. ProDOS can handle up to 64 directory levels, which translates out to an enormous number of filenames in lists if you wish to work your way through such a large number of file levels. The file structure used by ProDOS is called a hierarchical structure. This term is used to describe the use of differing levels of directories to store and manage files.

FILENAMES

DOS and ProDOS use different conventions for naming files. Under DOS 3.3, a filename may be up to 30 characters long. Under ProDOS, such names may only be 15 characters. DOS 3.3 filenames may contain spaces and all punctuation symbols except the comma. ProDOS filenames may not contain spaces and allow only the use of the period, in addition to numbers and letters. The filenames of both systems must begin with a letter but may contain numbers at any other point within the filename.

FILE TYPES

In DOS 3.3 only four types of files are commonly encountered by the user. These file types are identified by single-letter abbreviations when a catalog is displayed by using the CATALOG command. The file types encountered that are treated by abbreviations are Applesoft programs (A), binary files (B), text files (T), and Integer BASIC programs (I). ProDOS has the ability to use three-character symbols to identify 256 different file types for the user. Some of these file type abbreviations are user definable. See Chapter 5 for a list of common file-type abbreviations used by the *ProDOS User's Disk* and the Applesoft ProDOS commands.

MEMORY USE

DOS 3.3 and ProDOS use different sections of computer memory on the Apple //c computer. DOS 3.3 is always loaded into the upper 48K section of the computer's Random Access Memory(RAM). ProDOS loads into the upper section of the first 64K bank of memory on the Apple //c. This gives the user more space in the first 48K of RAM for use with programs. ProDOS does not support Integer BASIC and does not load it into the computer's memory.

When DOS 3.3 is booted, it automatically sets aside memory space to be used as a buffer for three open text files. The user planning to have more than three files open at once must use the MAXFILES command to set up space for the required number of files. ProDOS automatically opens buffer space as files are opened. This is a much

more efficient use of memory. Each ProDOS file buffer is 1024 bytes long as compared to file buffers of 524 bytes in DOS 3.3.

SUPPORTED HARDWARE

DOS 3.3 supports only the Apple Disk II disk drive or its equivalent, unless DOS is reconfigured or "patched" to handle hard disks or other larger-capacity storage devices. ProDOS was designed to handle Disk II disk drives or their equivalent, and to handle transactions with Apple Computer's hard disk system known as the Profile and any other device following the same protocol as the Profile. This device is not currently compatible with the Apple //c, but the Quark QC10 ten-megabyte hard disk is compatible with the Apple //c. This particular feature of ProDOS gives it the ability to assign as much as 32 megabytes of storage to a single volume and up to 16 megabytes of storage to a single file on a properly configured system.

OTHER CHARACTERISTICS

ProDOS communicates with a disk drive at a rate of eight kilobytes of data per second. DOS 3.3 communicates at a rate of one kilobyte per second. ProDOS will complete some disk operations up to eight times faster than DOS 3.3.

Storage devices are addressed by slot and drive number on DOS 3.3. They are addressed by volume names given to them by the user in ProDOS, but may also be referred to by slot and drive. These volume names may be changed as needed by the user in ProDOS. DOS 3.3 does not support interrupts, which allow a number of operations to be carried out at the same time. ProDOS supports interrupts.

A number of DOS 3.3 commands are not supported by ProDOS. These commands include FP, INT, INIT, MAXFILES, MON, and NOMON. ProDOS supports a number of commands that are not supported by DOS 3.3. These commands are CAT, CREATE, FLUSH, PREFIX, STORE, RESTORE, FRE, and -(DASH). A number of DOS 3.3 commands have been improved in ProDOS. These commands are APPEND, BLOAD, BRUN, BSAVE, CATALOG, CHAIN, CLOSE, IN#, PR#, OPEN, POSITION, READ, RUN, and WRITE. All of these differences and the nature of the improvements are discussed at the end of Chapter 7.

DOS vs ProDOS Command Table.

Operating System Commands (BASIC.SYSTEM Commands for ProDOS)	Description	DOS	ProDOS
APPEND	A sequential text file command that opens the specified file and writes the next data to the end of the current file. In ProDOS it may be used to append data to any type of file.	Yes	Yes
BLOAD	Instructs the computer to load a binary file. In ProDOS, it allows loading of the binary image of a file or part of a file.	Yes	Yes
BRUN	Instructs the computer to load and run a binary file from disk, or to run a binary file already stored in memory. In ProDOS, it may run a binary program file starting at any point in the file.	Yes	Yes
BSAVE	Instructs the computer to save a binary file from memory to disk. In ProDOS, it may save the binary image of any type of file except a directory file.	Yes	Yes
CAT	Causes ProDOS to display a 40-column catalog on the screen.	No	Yes
CATALOG	Causes DOS 3.3 to display a 40-column display of a disk directory on the screen. It causes ProDOS to display an 80-column display of a disk directory on the screen.	Yes	Yes
CHAIN	Runs a new program or file without clearing the old one from memory. In ProDOS, this command can chain one program to any line of another program. Does not work with Applesoft under DOS.	Yes	Yes
CLOSE	Closes one or more open sequential or random access text files. In ProDOS, files must be closed from within a program.	Yes	Yes

DOS vs ProDOS Command Table (continued).

Operating System Commands (BASIC.SYSTEM Commands for ProDOS)	Description	DOS	ProDOS
CREATE	A ProDOS command to create any type of file or a directory.	No	Yes
DELETE	Deletes files from the directory and clears the flags.	Yes	Yes
EXEC	Treats sequential text files as keyboard input.	Yes	Yes
FLUSH	Clears the file buffer in the computer's memory and transfers data onto a disk file.	No	Yes
FP	Accesses Applesoft BASIC from Integer BASIC.	Yes	No
FRE	Accesses the fast housekeeping commands of the operating system.	No	Yes
IN#	Opens the circuits to specified slots. In ProDOS, it may also be used to access machine language subroutines in memory as if they were programs available from a disk drive slot.	Yes	Yes
INIT	Formats disks.	Yes	No
INT	Accesses Integer BASIC from Applesoft BASIC.	Yes	No
LOAD	Loads programs from disk.	Yes	Yes
LOCK	Write-protects and delete-protects files on disk.	Yes	Yes
MAXFILES	Reserves a buffer for I/O.	Yes	No
MON	Displays disk I/O to the screen.	Yes	No
NOMON	Turns the disk I/O display to the screen off.	Yes	No
OPEN	Opens files and allocates a buffer. In ProDOS, this command may be used to open any type of file (not just a text file as in DOS 3.3).	Yes	Yes

DOS vs ProDOS Command Table (continued).

Operating System Commands (BASIC.SYSTEM Commands for ProDOS)	Description	DOS	ProDOS
POSITION	Reads or writes from a specified field of sequential text files and random access text files.	Yes	Yes
PREFIX	Sets a prefix from BASIC to make file handling more efficient.	No	Yes
PR#	Turns on the circuits to the specified slot. In ProDOS, it may be used to access specific memory addresses.	Yes	Yes
READ	Reads text files. In ProDOS, it allows a field number or byte number to be used as a starting place in the file, and READs any file type.	Yes	Yes
RENAME	Gives a file a new name on the directory.	Yes	Yes
RESTORE	Used to move variables from a file on a disk into a program that is running in the computer.	No	Yes
RUN	Loads and runs a program or runs a program that is already loaded. In ProDOS, line number may be specified from within another program as a starting place.	Yes	Yes
SAVE	Saves a file in memory to the disk.	Yes	Yes
STORE	Used to move number and string variables from the computer's memory to disk while the program is running. The file is given a .VAR attribute.	No	Yes
UNLOCK	Unprotects a file on disk.	Yes	Yes
VERIFY	Checks a file that was just copied or saved to verify that it matches the original. In ProDOS, it verifies that file is in the directory.	Yes	Yes

DOS vs ProDOS Command Table (continued).

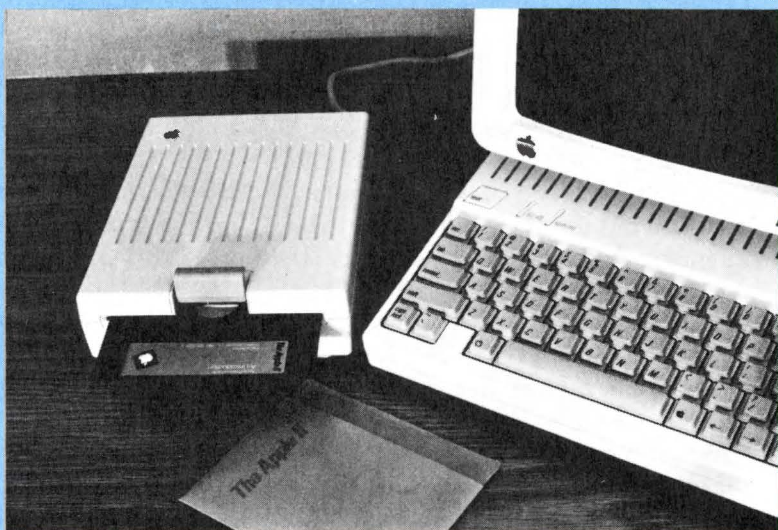
Operating System Commands (BASIC.SYSTEM Commands for ProDOS)	Description	DOS	ProDOS
WRITE	Writes data to an opened sequential or random access file. In ProDOS, WRITE may be specified to write a field or byte number and for any type of file except a directory.	Yes	Yes
- (dash)	Loads and runs a specified program from disk.	No	Yes

CONCLUSIONS

DOS 3.3 and ProDOS were compared generally in this chapter. The *System Utilities* disk that is supplied with the Apple //c contains a conversion program allowing the conversion of DOS 3.2 disks to DOS 3.3, the conversion of DOS 3.3 formatted disks to ProDOS format, and the conversion of ProDOS formatted disks to DOS 3.3 format. These features of the *System Utilities* disk make the purchase of a copy of DOS 3.3 unnecessary for most Apple //c users. ProDOS is more powerful and more convenient than DOS 3.3. It makes better use of memory and its commands are more powerful than those of DOS 3.3. A quick reference guide to ProDOS commands used from Applesoft is available on the fold-out reference card at the back of the book. If you want to snoop through the sectors, blocks, and bytes of a ProDOS disk to see how it is actually structured, use ProSNOOP from the *Apple //c User's Tools and Examples* disk sold in conjunction with this book.

7

The ProDOS User's Disk



ProDOS is the Apple //c's disk operating system. It also has the ability to handle interrupts and provides memory management. It performs a number of tasks required to take advantage of the convenience of mass data storage. It also maintains a directory of the files on each disk and manages those files that may consist of programs, data used by programs, text, or graphics.

ProDOS is the standard operating system for the Apple //c computer. An operating system is a systems program that enables a computer to interact with its peripheral devices and its mass data storage devices. Those operating systems that lean more heavily toward control of mass storage devices are called disk operating systems. ProDOS allows data to be transferred back and forth between the computer and a disk drive or a hard disk. ProDOS may also be used by the Apple II+ and Apple //e computers. ProDOS is supplied to the user on a 5¼-inch floppy disk. This disk must be inserted into the Apple //c's disk drive and be booted by turning on the computer, or by typing PR#6 followed by pressing the RETURN key.

This process causes the computer to read the ProDOS operating system from the system disk and to load ProDOS into the computer's memory. Once this process has been completed, the user has all of the features of ProDOS available for direct use or for use from within a program. On systems other than the Apple //c, ProDOS can be used to manage data that has been stored on Apple's Profile hard disk system. A hard disk system is being produced for the Apple //c. ProDOS can manage files on this system.

When you turn the Apple //c on with the *ProDOS User's Disk* in the disk drive (or press control- -reset), ProDOS is loaded into the computer's RAM memory. ProDOS then directs a sequence of events that eventually loads the *ProDOS User's Disk* utility program. This program allows you to do useful things such as formatting disks and copying data by making appropriate selections from a series of menus. It allows the copying of entire disks and the transfer of data from one disk to another. The process of copying entire disks can be done on the Apple //c with only the built-in disk drive, but this involves swapping the original for the copy several times during the process. It is more convenient and faster to use the internal disk drive for the original and an external disk drive for the copy.

The utility program on the *ProDOS User's Disk* translates the menu choices into appropriate directions to ProDOS to carry out the detailed steps required.

Now that the *ProDOS User's Disk* utility program has been introduced, the best way to learn what it does is to use it. If you already own the *ProDOS User's Disk*, insert it in the disk drive slot on the right hand side of the Apple //c. If you do not own the *ProDOS User's Disk*, you

can learn what it would be like to use it on your Apple //c by following along.

Turn on the computer. The disk drive will make some noise while the operating system is being loaded into the computer's memory. In a short time you will see the following screen display:

```
*****************************************
*                                       *
*           PRODOS USER'S DISK          *
*                                       *
*   COPYRIGHT APPLE COMPUTER, INC. 1983 *
*                                       *
*****************************************

YOUR OPTIONS ARE:

    ? - TUTOR: PRODOS EXPLANATION

    F - PRODOS FILER (UTILITIES)

    C - DOS <-> PRODOS CONVERSION

    S - DISPLAY SLOT ASSIGNMENTS

    T - DISPLAY/SET TIME

    B - APPLESOFT BASIC

PLEASE SELECT ONE OF THE ABOVE
```

This display is called The Main Menu. When a program displays a menu such as this one, it means that the user only needs to know what he or she wants to accomplish. There is no need to memorize and use a lot of strange words known as "commands." All that needs to be done is to press the letter key associated with the task the user wishes to perform. If you have never used ProDOS or any other operating system, press the question mark key. Pressing this key will cause the Apple //c to display a number of sequential screens that briefly explain operating systems in general and a few specifics about ProDOS.

Before proceeding further, let's examine The Main Menu in a little greater detail. The first choice after the Tutor is the ProDOS Filer. This program manages the files on a disk, identifies disks that are inserted into a disk drive, allows disks (volumes) to be named, prepares disks for storing data, allows disks to be copied, and allows files to be deleted from disks. These utilities are extremely important to the organization and management of data that is stored on disk or on a hard disk.

The next option is DOS <-> ProDOS Conversion. This program allows the user to convert files and programs that are already stored on disks formatted for DOS 3.3 for storage on disks formatted with ProDOS or vice versa. This feature will work with most public domain software and with any programs you may have written. A program or file that has been converted and stored in a new format may be run or used from the other operating system. It will probably not work with commercial software products, although it could convert files from commercial software if these files were normally stored on standard (not altered) DOS 3.3 formatted disks. Be sure to consult the software instructions or the manufacturer if you are attempting to convert commercial software to ProDOS.

The next option allows for the display of slot assignments. This option is designed primarily for the Apple II, Apple II+, and Apple //e as these computers are equipped with internal expansion slots for adding special processing cards, printer interface cards, modems, memory expansion cards, video display cards, and so on. It would only be used on the Apple //c to display memory available for data storage on the internal disk drive or an externally connected disk drive.

The next option is DISPLAY/SET TIME. This option requires a plug-in clock card or a built-in clock to operate automatically. Since the Apple //c has no built-in clock and no expansion slots for a clock card, the DISPLAY/SET TIME option may be used to manually set the date and time that ProDOS is being employed to record any updates to files or to programs that are made by the user. The time will remain fixed as set since there is no clock circuitry to update it.

The final choice on the menu is APPLESOFT BASIC. Choosing this option takes the user out of the *ProDOS User's Disk* Main Menu into Applesoft BASIC, with its conventions, syntax, and token words. ProDOS is still resident in memory and can be accessed through the ProDOS commands that are added to Applesoft BASIC by BASIC.SYSTEM. This language and its use will be dealt with in introductory terms in Chapter 8.

THE ProDOS FILER

Now that we have examined the *ProDOS User's Disk* Main Menu, let's try The Filer. This option on The Main Menu will probably be used more than any other option available to the Apple //c user. The Filer Menu is obtained by pressing the "F" key while The Main Menu is being displayed on the screen. The screen will clear and The Filer Menu will appear in its place. This menu is shown here:

```
****************************************
*                                      *
*    APPLE PRODOS SYSTEM UTILITIES     *
*          FILER VERSION 1.0           *
* COPYRIGHT APPLE COMPUTER, INC., 1983 *
*                                      *
****************************************

     ? - TUTOR

     F - FILE COMMANDS

     V - VOLUME COMMANDS

     D - CONFIGURATION DEFAULTS

     Q - QUIT

PLEASE SELECT AN OPTION
```

Pressing the question mark key causes a number of screen displays to appear. These displays explain The Filer and a number of technical terms associated with The Filer. Before going further, you should be familiar with some important concepts that are central to understanding and using ProDOS and the *ProDOS User's Disk* utility program properly. First, a disk must be prepared for use by ProDOS before it can accept data for storage from a computer that is using the ProDOS operating system. The process of readying a disk to store ProDOS files is called initialization or formatting. This process takes a short time and involves imprinting a disk with a directory, the starting and stopping places of specific blocks of memory, and the placement of various electronic signposts used by ProDOS to recognize where it is writing material to or reading material from.

Second, each floppy disk used by ProDOS is called a volume. Each volume must be given a distinctive name by the user. Files, which may consist of programs, text files, data, or graphics, may be stored on a floppy disk. These files are associated with a particular volume name by the computer as it uses ProDOS to manage the computer's files.

Third, pathnames are used by ProDOS to find a particular file on a particular disk volume. A pathname is used to specify the selection of a particular file from a particular volume, and even from a particular directory of files on a particular disk volume. A directory is a term that is used to specify a file containing the names, locations, and other information about a collection of files specified by the user.

Fourth, various wildcard characters may be used with the *ProDOS User's Disk* utility program as shortcuts to select and use files from a specified volume. These terms will be discussed in more detail later.

Volume commands are the first part of The Filer employed by most computer users. These commands and their use are presented below.

THE VOLUME COMMAND MENU

The first task to be done by a new user is to initialize a blank disk. To accomplish this task, insert a blank, uninitialized disk into the Apple //c disk drive. Then press the "V" key while The Filer Menu is being displayed. This action will cause the screen shown here to be displayed.

```
****************************************
*                                      *
*            VOLUME COMMANDS           *
*                                      *
****************************************

     ? - TUTOR

     F - FORMAT A VOLUME

     C - COPY A VOLUME

     L - LIST VOLUMES

     R - RENAME A VOLUME

     D - DETECT BAD BLOCKS

     B - BLOCK ALLOCATION

     K - COMPARE VOLUMES

 SELECT AN OPTION OR <ESC>:
```

Like previous menus, this menu has the TUTOR option. Press the question mark key if you need more information about volumes or volume commands. Otherwise, press the "F" key to instruct the computer to format the blank disk that is in the disk drive. As soon as this key is pressed, the program will instruct the computer to mark the beginning and end of each disk sector and write a directory file to the disk. This program divides the surface of the disk electromagnetically into 16 distinct one-sector blocks on 35 concentric tracks. Each of these blocks is capable of storing 256 bytes of data. Before the computer starts to format a volume, the user sees the following screen:

```
****************************************
*                                      *
*           FORMAT A VOLUME            *
*                                      *
****************************************

- FORMAT -
   THE VOLUME IN SLOT: (6)
                 DRIVE:

   NEW VOLUME NAME:

   PRESS (RET) TO ACCEPT:<ESC> TO EXIT-
```

The blanks should be completed by entering a "6" at the cursor location, which requests the slot number of the blank disk that will be initialized. The drive number should be entered next if your system uses two disk drives to assure that the disk being initialized is the disk you intend to initialize. The program will automatically insert a "1" in the blank space. This "1" is a default value that was written into the program by the programmer. It is a common time-saver in many programs. On a two-disk drive system, you would enter the number "2" into this blank if the disk you wished to initialize had been inserted in the external disk drive.

As soon as the slot and drive have been specified, the program requests that a name be given to the disk being initialized. This name must be no longer than 15 total characters. It must start with a letter and it may contain numbers, but it may not contain spaces. The only punctuation mark allowed in a volume name is a period. Each volume name must be preceded by a slash (/). If you planned to save a disk full of games written by you or your friends, you could name the volume being initialized with a name such as /GAMES1. If you did not want to name this volume but still wanted to initialize it, you would press the RETURN key when the volume name was requested. This would cause the computer to give the volume the name /BLANK followed by a two-digit number between 00 and 99. The program remembers the last number used and applies the next sequential number immediately after the name BLANK. Later, when you have decided on a name, you can use the Rename a Volume option on the menu to do so.

If you inadvertently placed an initialized disk called GAMES2 in the disk drive and pressed the "F" key, the program asks 'DESTROY 'GAMES2'? (Y/N). If you want to overwrite the contents of this disk and destroy any files on it, you would press the "Y" key and the disk would be initialized. If not, you would press the "N" key, remove the initialized disk, and replace it with a blank, uninitialized disk. If you

have changed your mind and do not wish to initialize a disk, you would press the ESC key to return to the Volume Command Menu.

Once all of the steps described above have been completed to the user's satisfaction, the disk drive will start running. The computer will name the volume with the name selected by the user, write the directory file, and mark the sectors on the disk. The disk will then be usable by the ProDOS operating system.

COPYING VOLUMES

The next option on the Volume Command Menu is C - COPY A VOLUME. This option allows the user to copy the contents of an entire disk to another disk. Unless you wish to overwrite the contents of the copy disk with the contents of the original, you should use blank, uninitialized disks for copies. It is a good idea to make a copy (backup) of any disks that contain programs or files that you consider important.

One of the most important disks that you can back up is the *ProDOS User's Disk*. You should make a copy of this disk as soon as you have a blank disk available to do so. Before you start to copy this valuable disk, place a write-protect tab over the notch on the side of the diskette. This is a very important step to prevent losing your *ProDOS User's Disk* (or any valuable disk you copy). For more on this, see the instructions for copying the *System Utilities* in Chapter 5. To copy this disk, insert the *ProDOS User's Disk* in the disk drive on the Apple //c. Then select the Copy a Volume option from the Volume Command Menu by pressing the letter "C". After this key has been pressed, the following screen will be displayed:

```
****************************************
*                                      *
*             COPY A VOLUME            *
*                                      *
****************************************

- COPY -
   THE VOLUME IN SLOT: (6)
                DRIVE:

    TO VOLUME IN SLOT:
                DRIVE:

       NEW VOLUME NAME:

 PRESS <RET> TO ACCEPT:<ESC> TO EXIT -
```

When you first see this menu, a "6" will be displayed in the parentheses at the first cursor location after the words "THE VOLUME IN SLOT:". This value is the value considered by the person who wrote the program as the most likely value the user would enter. This value is called a default value. It provides a shortcut to the user and also suggests the type of answer required by the system. If the disk to be copied is in the built-in disk drive on the Apple //c, you would press the RETURN key to direct the program to accept the value "6" in the first cursor position. This action will cause the cursor to appear at the second position, which requests that the disk drive that contains the original be specified by the user. A "1" will appear as a default value at this cursor position.

If the original disk is in the built-in disk drive, you would once again press the RETURN key to accept the default value. The cursor then jumps to the third location where the slot of the disk drive containing the copy disk is requested. If you only have the built-in disk drive on the Apple //c, you would press the RETURN key at this time. This instructs the system to accept the default value of "6". The cursor then jumps to the next screen location, which asks you to indicate the disk drive number of the disk drive that will contain the copy. If you only have the internal disk drive on your Apple //c, you would insert the number "1" at this cursor location and then press the RETURN key. If you have a second, external disk drive attached to the computer and the copy disk is in this disk drive, you would accept the default value of "2" at this cursor location by pressing the RETURN key.

Once this step has been completed, the cursor will move to the location where the name of the new volume is requested. This volume could be named USERS.DISK or any name that describes its contents. When the name has been entered and the RETURN key has been pressed, the disk drive will begin to run and the contents of the USERS DISK will be transferred to the computer's memory until it is full.

If you only have the built-in disk drive, the disk will stop turning and a message will appear on the screen requesting that the original disk be taken out of the disk drive, and be replaced by the copy disk. Once this has been done, the disk will be initialized and the contents of the computer's memory will be transferred to the copy disk. The computer will then ask that the copy disk be removed and the original disk be placed in the disk drive. The computer will then begin reading the contents of the disk into its memory at the point where it left off when the memory became full before. If the memory again becomes full or the entire contents of the original disk have been transferred into memory, the computer will once again instruct the user to remove the original from the disk drive, replace it with the

copy disk, and press RETURN or another key. Once the copy has been completed, the program will ask that the copy be removed and the original be placed in the disk drive.

If you have a second, external disk drive connected to the Apple //c, the disk swapping steps described above will not be required and the process of copying the volume will be carried out more quickly than they would be carried out as described above.

LIST VOLUMES

The next command available from the Volume Command Menu is the command to list volumes. This command is executed when you press the "L" key while the Volume Command Menu is being displayed on the screen. When you press this key, the computer will read the name of each volume currently in a disk drive and will display the volume names in a list on the screen. If you had placed GAMES1 in the built-in disk drive and USERS.DISK in the external disk drive, the following display would appear on the screen:

```
****************************************
*                                      *
*             LIST VOLUMES             *
*                                      *
****************************************

    SLOT  DRIVE      VOLUME NAME

     3      2      /RAM

     6      1      /GAMES1

     6      2      /USERS.DISK


--PRESS <RET> TO BEGIN: <ESC> TO EXIT--
```

If you pressed the RETURN key at this time, the program would run again. If no disk is in either disk drive, only the /RAM name will appear on the list shown above. If a non-ProDOS disk is in either disk drive, the message <NO DIRECTORY> will appear on the list in place of a volume name for the slot and drive containing the non-ProDOS disk. The volume name /RAM would appear on the slot 3 drive 2 line.

The /RAM volume name refers to the 64K of memory expansion that is on the extended 80-column card of the Apple //e. It is built into the Apple //c's memory, but it is treated by the computer and ProDOS as if it were on an expansion card. ProDOS treats this extra memory space as if it were another disk drive. Programs may be copied into this area and used as if they were on a disk drive. This is accomplished by using the COPY FILES command. Access time will be much faster than the time required to access a normal disk drive. Programs and files that have been stored in this memory space may by copied to a regular disk drive by using the COPY FILES command of the FILE MENU. This volume may be renamed with any legal volume name you desire.

RENAMING VOLUMES

The next option on the menu is R - RENAME VOLUMES. This option allows the user to change a volume name whenever he or she needs to make such a change. To change the name of any volume, insert the disk containing the volume into one of the disk drives. Press the "R" key while the Volume Command Menu is being displayed. The following display will appear on the screen:

```
****************************************
*                                      *
*           RENAME A VOLUME            *
*                                      *
****************************************

--RENAME--
   THE VOLUME IN SLOT: (6)
                DRIVE:


NEW VOLUME NAME:




--PRESS <RET> TO ACCEPT: <ESC> TO EXIT--
```

Enter the slot number of the disk drive containing the volume to be renamed at the first cursor position. For the Apple //c's internal or external disk drive, this slot number would be a "6". Enter the

disk drive number at the next cursor position. For the internal disk drive, this would be a "1". For the external disk drive, it would be a "2". Enter the new volume name at the next cursor position and press the RETURN key to accept this new name and to instruct the computer to record this name on the disk. If you made an error in the slot or drive specification, or if you made a mistake in the new name, you would tell the computer to abort the renaming process by pressing the ESC key. When the renaming process is complete, the message RENAME COMPLETE will be printed on the screen by the program.

DETECT BAD BLOCKS

If you suspect that a disk is bad, damaged, or has accepted the copy of the contents of another disk improperly, you can check the surface of the disk and the quality of the copy by using the DETECT BAD BLOCKS command on the Volume Command Menu. In the case of improperly copied data, the problem may or may not be detectable. The disk to be checked must be an initialized disk. While the Volume Command Menu is being displayed, press the "D" key. The following display will be printed on the screen by the program:

```
****************************************
*                                      *
*           DETECT BAD BLOCKS          *
*                                      *
****************************************

--DETECT BAD BLOCKS--
  FOR VOLUME IN SLOT: (6)
               DRIVE:










--PRESS <RET> TO ACCEPT: <ESC> TO EXIT--
```

The default value of "6" should be accepted on the Apple //c by pressing the RETURN key while the cursor is at its first screen position. If the disk to be checked is in the internal disk drive, accept the default value of "1" by pressing the RETURN key at the second cur-

sor position. If the disk is in the external disk drive, enter a "2" at this position and press the RETURN key. The computer will check all of the blocks that have been set up to receive data on the disk. It will not erase any data already stored on the disk. If no errors are detected, the message "0 BAD BLOCKS" will be printed on screen. If any bad blocks are found, a message such as the one shown here will appear on screen:

```
BAD BLOCK NUMBER:

   2
  12
  15
```

When a disk is initialized, blocks are numbered sequentially from 0 to 279. The numbers displayed above are the block numbers of the bad blocks found by the program.

If you discover a number of bad blocks of data on any of your disks, you should try to copy the files from that disk one at a time. As each file or program is transferred, try it. Those data files that are incomplete or those programs that will not copy correctly are lost unless you have a complete knowledge of data files, file handling, and the ProDOS operating system. Even then you will be fortunate to recover a lost file or program. A number of file recovery programs and ProDOS utility programs are available. One of these programs might help you recover lost files.

BLOCK ALLOCATION

The next option on the Volume Command Menu is the Block Allocation option. This option is selected by pressing the "B" key while the Volume Command Menu is on the screen. The Block Allocation option is used to determine how much space on a disk is occupied, how much space is available, and how many total blocks are available for use. This option is extremely useful when you are unsure of whether a program or a data file will fit on a partially used disk. When the "B" key is pressed, the following display is printed on the screen:

```
****************************************
*                                      *
*           BLOCK ALLOCATION           *
*                                      *
****************************************

-BLOCK ALLOCATION-
  FOR VOLUME IN SLOT: (6)
               DRIVE:









-PRESS <RET> TO ACCEPT: <ESC> TO EXIT -
```

As you did previously, press the RETURN key to accept the default value of "6" if the disk to be checked is in the internal disk drive of the Apple //c, or if it is in the external drive plugged into the rear panel of the computer. Enter the number "1" at the next cursor position if the disk is in the internal disk drive. Enter the number "2" if it is in the external disk drive. The computer will then read the disk in the specified drive and display a screen similar to this:

```
****************************************
*                                      *
*           BLOCK ALLOCATION           *
*                                      *
****************************************

-BLOCK ALLOCATION-
  FOR VOLUME IN SLOT: (6)
               DRIVE: 1

           120 BLOCKS USED

           160 BLOCKS FREE

           280 BLOCKS TOTAL



-PRESS <RET> TO ACCEPT: <ESC> TO EXIT-
```

If no such message appears on the screen, you will see an error message and should consult your ProDOS manual to determine what that message means and what to do about it. A newly formatted disk will show:

```
****************************************
*                                      *
*           BLOCK ALLOCATION           *
*                                      *
****************************************


-BLOCK ALLOCATION-
  FOR VOLUME IN SLOT: (6)
               DRIVE:  1

             7 BLOCKS USED

           273 BLOCKS FREE

           280 BLOCKS TOTAL


-PRESS <RET> TO ACCEPT: <ESC> TO EXIT-
```

The seven used blocks are directory and ProDOS reserved areas. (Technically, reserved areas are for the boot code used when the power is turned on or the computer is cold started, and for the bit map of the disk.)

COMPARE VOLUMES

The final command on the Volume Command Menu is C - COMPARE VOLUMES. This command allows the user to determine how successful a COPY VOLUME command has worked. It compares an original with a copy to determine whether they are exactly the same. If they are not the same, the program displays the numbers of those blocks that are not identical. This program may be used with either a one- or two-disk drive system. It will operate much faster on a two-disk drive system. This option is selected by pressing the "C" key while the Volume Command Menu is being displayed. It will produce the following screen display

```
****************************************
*                                      *
*            COMPARE VOLUMES           *
*                                      *
****************************************

-COMPARE-
  THE VOLUME IN SLOT: (6)
               DRIVE:

   TO VOLUME IN SLOT:
               DRIVE:




-PRESS <RET> TO ACCEPT: <ESC> TO EXIT-
```

If you are using only the internal disk drive on the Apple //c, you would accept the default value of "6" at the first cursor position, and the default value of "1" at the second cursor position by pressing the RETURN key. You would then enter a "6" in the next cursor position and a "1" in the last cursor position. The program will tell you when to swap disks as the volumes are compared.

With a two-drive system, you would enter a "2" in the final cursor position. All other entries would be the same as those used with a single-drive system. This command may also be used to compare two DOS 3.3 disks. If the comparison indicates that all of the blocks are identical, the message "COMPARE COMPLETE" will be displayed. If a number of blocks do not match, a message similar to this one appears on the screen:

```
BLOCK NUMBERS DO NOT MATCH:

          2

          8

-PRESS <RET> TO CONTINUE: <ESC> TO EXIT-
```

Some copy-protected programs will "fool" the FILER. It will think the copy is the same but there will be undetectable differences that prevent the copy from working.

The descriptions of the Volume Commands and how to use each one should help you realize the full potential of the ProDOS User's

Disk. Next, we will discuss the File Command Menu included in the Filer Menu accessed from the *ProDOS User's Disk* Main Menu.

FILES AND FILE TERMS

The second option on the Filer Menu is F - FILE COMMANDS. This option allows the Apple //c user to manage individual files and programs stored on a volume. Before discussing each of the commands available from this menu, the user should be familiar with a number of important terms. A file is a specific collection of data generated by a program, a word processor, a spreadsheet program, a text editor, or other data. It may also be a program that has been stored on a disk.

Each file must be given a filename by the user. A filename must begin with a letter and contain no more than 15 characters. It may include numbers and periods but it may not have spaces or other punctuation marks.

The ProDOS file structure is a hierarchical structure. This means that files related by some common characteristic or files that the user wishes to associate in some fashion may be classified in directories or subdirectories.

A directory for a volume lists all of the files on that volume. This directory is one of the files written to the volume when a disk is initialized. It is updated each time a new file is stored on a disk or an old file is deleted from a disk. Each volume directory is limited to a total of 51 filenames. When you have stored this number of files on a disk, the variety may be somewhat overwhelming. It is only logical that you should be able to classify like files and list them under a subdirectory which then becomes a single listing in a volume directory.

Commands are available on the File Command Menu to list both volume directories and subdirectories. If you wish to copy, read, delete, open, close, or write to a particular file on a disk, you must be able to instruct the computer to access the file. This instruction takes the form of a command followed by a pathname on an Apple //c that uses ProDOS as an operating system.

A pathname is a sequence of names that tells the computer how to find the particular file the user wishes to access. It must include the volume name and filename desired, and if the file has been included in one or more layers of subdirectories, each subdirectory name must be included prior to providing the filename. Parts of pathnames, such as the volume name, directory name, and subdirectory name, may be associated as a prefix, with the "Set Prefix" command. ProDOS then automatically inserts the contents of the prefix when a filename is entered without any slash. Such instances occur when you are instructing the program to list, copy, delete, compare,

or alter write-protection for a particular file. The process of associating volume directory names and subdirectory names with filenames is called setting the prefix, which is a command available from the FILE COMMAND MENU.

When you are using the FILER's file commands to manage the files on volumes, you may use one of two wildcard characters with most commands. The first wildcard is the equal sign (=), which instructs the program to copy, compare, or delete all files with the same name or files of the same type (provided the file type has been designated by using a period followed by a three-letter abbreviation at the end of the filename). The second wildcard character is the question mark (?). This character is used with list, copy, compare, and delete commands requiring the system to display the name of each file of the same name or same type as specified by the user prior to taking any action with the file. Now that we are familiar with the terminology associated with files, and the treatment of files when you are using the File Command Menu, in the FILER program on the *ProDOS User's Disk*, it is time to use the File Command Menu.

FILE COMMANDS

The File Command Menu is obtained by selecting the letter "F" when the ProDOS FILER (UTILITIES) Menu is displayed on screen. This menu looks like this:

```
****************************************
*                                      *
*             FILE COMMANDS            *
*                                      *
****************************************
          ? - TUTOR

          L - LIST PRODOS DIRECTORY

          C - COPY FILES

          K - COMPARE FILES

          A - ALTER WRITE PROTECTION

          R - RENAME FILES

          M - MAKE DIRECTORY

          P - SET PREFIX
SELECT AN OPTION OR <ESC>:
```

Before using the features of this directory, let's briefly examine each command. The TUTOR command works the same as it did on the Volume Command Menu. The LIST PRODOS DIRECTORY command is used to list all of the filenames on the volume. The COPY FILES command is used to copy files from one volume to another. The DELETE FILES command is used to remove specified files from a disk. This command should only be used when you are certain that you wish to remove a file. The file is lost once this command has been used. The COMPARE FILES command is used to compare the original of a file that has been copied with the copy of the file to ensure that the copy was made successfully. The ALTER WRITE-PROTECTION command is used to lock files to prevent the system from writing further data to the file, and to unlock files to allow writing data to them. The RENAME FILES command allows filenames to be changed. The MAKE DIRECTORY command is used to create a file in a volume directory or a subdirectory, which is used as a subdirectory. The SET PREFIX command is used to specify a volume name and possibly a subdirectory name when an instruction is being given. Now that we are familiar with the File Commands, let's use them.

LIST ProDOS DIRECTORY

Make sure the *ProDOS User's Disk* is in the internal disk drive of the Apple //c. With the FILE COMMANDS Menu displayed, press the "L" key. This will cause the display shown here to appear on the screen:

```
****************************************
*                                      *
*         LIST PRODOS DIRECTORY        *
*                                      *
**PREFIX: /USERS.DISK/ *****************


--DIRECTORY--
   PATHNAME: (                         )






-- ENTER PATHNAME AND PRESS <RET>--
```

If you press the equal sign at this time and then press RETURN, the following display will appear on the screen:

```
DIRECTORY:/USER.DISK/

 NAME                TYP   BLOCKS   MODIFIED

*PRODOS              SYS       31    1-JAN-84
*BASIC.SYSTEM        SYS       21   15-NOV-83
*FILER               SYS       51   29-MAR-84
*CONVERT             SYS       42    1-NOV-83
*STARTUP             BAS        3   15-OCT-83
*HYPNOSIS            BAS        3   15-OCT-83
*ANIMALS             BAS       10   15-OCT-83

 BLOCKS FREE      88       USED:      192

 --PRESS <RET> TO BEGIN: <ESC> TO EXIT--
```

This display provides the key facts about the files stored on the *ProDOS User's Disk*. A similar display can be produced for any disk that is the named volume in the PREFIX section of the display. The asterisk prior to the filename signifies that the file is locked and may not be written to. The file type is displayed after the filename. File types are identified by the abbreviations listed here:

Abbreviation	**Description**
SYS	A system file that is required to use ProDOS.
BAS	A program written in Applesoft BASIC.
TXT	A text file written by a program.
$00	A file of no known type.
BAD	Bad Blocks not available to store.
TXT	A text file in ASCII code.
BIN	A Binary file (machine language program), a hi-res picture, data.
DIR	A Directory file.
VAR	A Variable file for an Applesoft BASIC program.
REL	A Relocatable code file.

This list does not include all of the possible file type abbreviations used by ProDOS, but it does list those that you are most likely to encounter. The next section of the display indicates the number of 512 byte blocks used by each file. The next portion of the listing shows when each file was last written to or modified. To ensure that this notation is made, you should use the DISPLAY/SET TIME option of the Main Menu to set the current date each time you boot ProDOS. The lower portion of the display shows how many of the

available storage blocks are used and how many are free. Press the ESC key to get back to the File Commands Menu. Make sure the *ProDOS User's Disk* is in the computer's internal drive. If you have an external disk drive, place a backup copy of the EXAMPLES disk without a write-protect tab in this disk drive. We will then copy a file from one volume to another.

COPY FILES

The COPY FILES command is obtained by pressing the "C" key when the File Commands Menu is being displayed on the screen. The following display is printed to the screen when this command is called from the menu:

```
****************************************
*                                      *
*              COPY FILES              *
*                                      *
**PREFIX: /USERS.DISK/ *****************

-COPY-
   PATHNAME: (                         )


     TO PATHNAME:







-ENTER PATHNAME AND PRESS <RET>-
```

Make sure the *ProDOS User's Disk* is in the internal disk drive and a backup copy of the EXAMPLES disk without a write-protect tab is in the external disk drive. If you do not have an external drive, remove the *ProDOS User's Disk* and place the EXAMPLES disk in the disk drive when you are directed to do so. Enter /USERS.DISK/ANIMALS at the first cursor position and press the RETURN key. Enter /EXAMPLES/PROGRAMS/ANIMALS at the second cursor position and press the RETURN key. This will copy the program ANIMALS from the *ProDOS User's Disk* to the EXAMPLES disk. Now list the ProDOS Directory for /EXAMPLES/PROGRAMS. This will display all of the files in the PROGRAMS subdirectory. ANIMALS will be there if you have had no problems with the copy

sequence. This file is locked, which is signified by the asterisk beside the filename. To delete or change this file you would first be required to alter its write-protection.

ALTERING WRITE-PROTECTION

To alter the write-protection for any file, you press the "A" key while the File Command Menu of the ProDOS Filer Menu is being displayed. Let's alter the write-protection of the ANIMALS program that we have just copied to the EXAMPLES disk. When you press the "A" key, the following display appears on the screen:

```
*****************************************
*                                       *
*        ALTER WRITE-PROTECTION         *
*                                       *
**PREFIX: USERS.DISK/********************


-ALTER WRITE-PROTECTION-
   PATHNAME: (                          )







-ENTER PATHNAME AND PRESS <RET>-
```

Enter /EXAMPLES/PROGRAMS/ANIMALS at the cursor position and press RETURN. This instructs the computer to access the file called ANIMALS from the PROGRAMS directory on the EXAMPLES disk rather than the same program on the USER.DISK, which is the default volume assumed by ProDOS. The internal disk drive of the Apple //c is assumed to be the default disk drive by ProDOS. The screen will then display the question "LOCK FILES? (Y/N)." To unlock a locked file you would respond by pressing the "N" key. After you have done this for the ANIMALS program on the EXAMPLES disk, you can delete this program from the disk. Before trying to delete this file, let's compare it with the original to see if it was copied correctly.

COMPARE FILES

The COMPARE FILES command is obtained by pressing the "K" key while the File Commands Menu is on the screen. Make sure that the *ProDOS User's Disk* is in the internal disk drive of the Apple //c and the EXAMPLES disk is in the external disk drive. If you only have the internal disk drive, make sure the *ProDOS User's Disk* is in the disk drive. Make your way to the File Commands Menu and press the "K" key. The following screen will appear:

```
****************************************
*                                      *
*             COMPARE FILES            *
*                                      *
**PREFIX: /USERS.DISK/******************

-COMPARE-
   PATHNAME: (                         )





     TO PATHNAME:









-ENTER PATHNAME AND PRESS <RET>-
```

Enter /USERS.DISK/ANIMALS at the first cursor position. Enter /EXAMPLES/PROGRAMS/ANIMALS at the second cursor position and press RETURN. If you only have the internal disk drive, the computer will read the file from the *ProDOS User's Disk* and then instruct you to insert the DESTINATION DISK. Take the *ProDOS User's Disk* out of the disk drive at this time and insert the EXAMPLES DISK. The computer will then compare the two files. If the two files are identical, the message COMPARE COMPLETE will be printed on the display screen. If the two files are not identical, the message FILES DO NOT MATCH will be printed to the screen. If this happens, recopy the files until the Compare Files program indi-

cates that the files match. Keep in mind that it is possible that a file successfully copied from a protected disk may not function correctly on another disk.

If you have an external disk drive, insert the *ProDOS User's Disk* in the internal disk drive and the EXAMPLES disk in the external disk drive. Follow the procedures outlined above. The computer will first read the original file from the internal disk drive, then it will compare the file on the copy disk. The message COMPARE COMPLETE will appear in the case of a successful copy. The message FILES DO NOT MATCH will appear in the case of an unsuccessful copy. Before deleting the extra copy of ANIMALS on the EXAMPLES disk, let's rename the file as GUESSING.GAME.

RENAMING FILES

The Files Command Menu contains the RENAME FILES option. This option is selected by pressing the "R" key while this menu is on the screen. This option may be used any time you wish to change a filename to prevent confusion or for any other purpose. It will not change the name of a locked file or a file on a write-protected disk (with tabs in place). When you press the "R" key, the following screen is displayed:

```
****************************************
*                                      *
*             RENAME FILES             *
*                                      *
****************************************

-RENAME-
   PATHNAME: (                         )



   NEW PATHNAME:








-ENTER PATHNAME AND PRESS <RET>-
```

To change the name of the program ANIMALS to GUESSING.GAME, you would enter /EXAMPLES/PROGRAMS/ANIMALS

at the first cursor position and press the RETURN key. The cursor would jump to its next position (NEW PATHNAME:) where you would enter /EXAMPLES/PROGRAMS/GUESSING.GAME. If the file is renamed successfully, the message RENAME COMPLETE will be printed on the screen. If there is another file on the volume with the same name, the message DUPLICATE FILENAME would be printed on the screen. Now that we have dealt with the LIST, COPY, COMPARE, and RENAME commands on the File Command Menu, we can finally delete the extra copy of the ANIMALS program. Do not forget that it has been renamed GUESSING.GAME.

DELETING FILES

Files may be deleted from any volume (that is, not write-protected with a write-protect tab) by using the DELETE FILES command from the File Command Menu. Locked files, however, cannot be deleted unless they are unlocked. This command is used by typing the letter "D" when the File Command Menu is on the screen. When this command is selected, the following display is printed on the screen:

```
****************************************
*                                      *
*             DELETE FILES             *
*                                      *
****************************************

-DELETE-
   PATHNAME: (                         )








-ENTER PATHNAME AND PRESS <RET>-
```

To delete the file GUESSING.GAME from the EXAMPLES disk, enter /EXAMPLES/PROGRAMS/GUESSING.GAME at the cursor position and then press the RETURN key. The computer then causes the file to be deleted from the volume and the message DELETE COMPLETE will be printed to the screen. You could not delete the directory PROGRAMS or any other directory on a disk unless all the programs in the directory had been moved to another directory or

had been unlocked and deleted. A directory must be empty of all associated files before it can be deleted.

All files with similar names can be deleted at the same time by using the equal sign wild card. If you had two files in the PROGRAMS directory named ARCADE.GAME and ADVENTURE.GAME, you could delete both files at once by entering /EXAMPLES/PROGRAMS/=GAME. This use of the wildcard character would cause all of the files in the PROGRAMS directory with an ending of GAME to be deleted. If you used the question mark wildcard with the DELETE FILES command, the program would ask for a Y/N response before a file was deleted.

MAKING DIRECTORIES

The next option on the File Command Menu is the MAKE DIRECTORY command. This command is used by pressing the "M" key when this menu is on the screen. The directory must be present on the disk before files can be included in the directory by copying or by saving. For example, if you had formatted a disk called /BACKUP to backup a few of the files on EXAMPLES, you would set the subdirectory names prior to copying the files onto this disk. Once the volume /BACKUP has been formatted, you could get to the File Command Menu and press the "M" key to obtain the display shown here:

```
****************************************
*                                      *
*            MAKE DIRECTORY            *
*                                      *
****************************************

-MAKE DIRECTORY-
   PATHNAME: (                         )










-ENTER PATHNAME AND PRESS <RET>-
```

To set up a directory for /GAMES, enter /BACKUP/GAMES at the cursor position and then press RETURN. To set up another directory

called /PROGRAMS, repeat the process by entering /BACKUP/PROGRAMS at the cursor position. This sets up a directory for GAMES and another for PROGRAMS on the volume named BACKUP. To place programs in these directories, copy the appropriate programs or files from the EXAMPLES disk, any other disk you may desire, or from the memory of the computer. For instance, if you wanted the program called SELFLIST included in the PROGRAMS directory on the BACKUP disk, run the COPY FILES option discussed above and enter /BACKUP/PROGRAMS/SELFLIST at the cursor position that requests the destination pathname. This ensures that the program SELFLIST is included in the PROGRAMS directory.

SET PREFIX

The SET PREFIX option is the last option on the File Command Menu. It offers a shortcut when you are going to copy, list, compare, or delete a number of files from the same source or the same destination. This may be a volume or a subdirectory. To use this command, press the "S" key when the File Command Menu is being displayed but also before employing the other commands. For instance, if you wanted to avoid typing the key sequence /EXAMPLES/ before listing the directories on the EXAMPLES disk, you would set the prefix for the list command by pressing the "S" key to use the screen shown here:

```
****************************************
*                                      *
*              SET PREFIX              *
*                                      *
****************************************

--SET PREFIX--
  NEW PREFIX: (                        )








-ENTER PATHNAME AND PRESS <RET>-
```

Enter /EXAMPLES followed by pressing the RETURN key at the cursor position. This alters the bottom line of asterisks on the SET PREFIX screen header and all other File Command screen headers to display the Prefix /EXAMPLES. To use this feature with the LIST PRODOS DIRECTORY command, you would access the command in the fashion described earlier. Then enter PROGRAMS at the cursor location requesting PATHNAME. The /EXAMPLES/ portion of the pathname would automatically be entered by the program to take advantage of the default prefix you set with the SET PREFIX command. The program would then list all of the programs in the PROGRAMS subdirectory to the screen. This command really helps if you are going to perform repetitive tasks with a file command. ProDOS must be able to find the volume directory and all subdirectories named in the prefix or it will not be accepted.

CONFIGURATION DEFAULTS

Now that we have examined the commands that make up the Volume Command Menu and the File Command Menu, let's return to the ProDOS Filer Menu for the last option on the menu. The Filer Menu contains an option that allows the system configuration to be defined to set up the origin and destination disk drives for making copies, and to set up the output of the program to the screen or to the screen and a printer at the same time. This option is obtained by pressing the "C" key when the Filer Menu is being displayed. This action produces the following display:

```
****************************************
*                                      *
*        CONFIGURATION DEFAULTS        *
*                                      *
****************************************

         ? - TUTOR

         S - SELECT DEFAULTS

         R - RESTORE DEFAULTS

SELECT AN OPTION OR <ESC>:
```

The tutor is self-explanatory. The Select Defaults option presents a screen allowing the user to designate which disk drive is to be used for originals and which is to be used for copies as a default. It also allows the user to tell the system to display activity on the screen

only or to a printer and the screen at the same time. The Restore Defaults option restores the original default values set by programmers. These original default values are slot 6, drive 1 for the origin drive; slot 6, drive 2 for the destination drive; and "screen only" for the display of directory and volume listings, and so forth. All of the subsequent screens used by this command are self-explanatory and have not been included in this book. If you have only one drive, it would make sense to set the default for the destination drive to slot 6, drive 1 (the internal drive).

DOS <-> ProDOS CONVERSION

One of the most important programs available on the *ProDOS User's Disk* is the DOS <-> ProDOS CONVERSION program. This will be the case until ProDOS becomes the de facto standard operating system for all Apple computers of the Apple II family. ProDOS is currently being supplied with all new Disk II disk drives, and all new Apple //e and Apple //c computers. The *System Utilities* disk is the ProDOS disk included with the Apple //c. Software manufacturers have been working with ProDOS since the middle of 1983 and have converted many existing programs to work with ProDOS. Most new applications have been written for ProDOS.

All of these factors make it necessary for users to have the ability of converting files from one operating system to the other. The conversion program allows data, files, and programs that have been stored on disks formatted by DOS 3.3 to be converted and stored on disks formatted by ProDOS and vice versa. ProDOS cannot read files from a disk that was formatted by DOS 3.3 and cannot write files to such disks. (*System Utilities*, however, can copy files from either DOS 3.3 or ProDOS disks.) The converse is true of DOS 3.3. Therefore, the *ProDOS User's Disk* includes a program for the conversion of files from one operating system to the other with some limitations. Files written by a program such as a word processor that operates with DOS 3.3 and is saved on a DOS 3.3 formatted disk cannot be used by a ProDOS version of the same program. Files generated by such programs may be converted from one operating system to the other to allow free use by either the DOS 3.3 or ProDOS version of the program.

Programs that you have written and stored on DOS 3.3 disks, and most public domain programs stored on DOS 3.3 formatted disks, may also be converted to the ProDOS format. (Remember that the *System Utilities* disk also provides conversions from DOS 3.2.) Many text files, data files, and most binary files may be converted. Ran-

dom access files may not be converted and some Applesoft programs may need to be modified before they can be used on the other operating system. Most binary programs will probably not work under ProDOS until a new ProDOS version is available. Very few commercial software programs will be convertible. Be sure to read the documentation that accompanies any commercial software you buy to determine which operating system it uses before using the software. If there is any doubt, call the manufacturer. The DOS <-> ProDOS CONVERSION program is easy to use. Press the "C" key while the ProDOS Main Menu is being displayed onscreen. This action will cause the screen to display the following:

```
               CONVERT Menu
Direction: DOS 3.3 S6,D2 ---> ProDOS
Date: <NO DATE>
Prefix: /USERS DISK/

---------------------------------------

   R - Reverse Direction of Transfer

   C - Change DOS 3.3 Slot and Drive

   D - Set ProDOS Date

   P - Set ProDOS Prefix

   T - Transfer (or List) Files


---------------------------------------

Enter Command: ?   ? - Tutor,  Q - Quit
```

This menu makes all of the features of the DOS <-> ProDOS conversion program available to the ProDOS user. The direction of transfer is established in the header section of the menu. As illustrated in the display shown above, it is assumed that the disk containing DOS 3.3 files for conversion to ProDOS is in the external disk drive of the Apple //c. It also assumes that a ProDOS formatted disk is in the internal disk drive. For the purpose of learning how to use this conversion program, place a copy of your DOS 3.3 *System Master* disk in the external disk drive. If you have only one drive, see, "Converting with One Disk Drive" below. Remove the *ProDOS User's Disk* from the internal disk drive and replace it with any ProDOS formatted disk with enough room to hold two or three additional programs. Press the "P" key to set the ProDOS Prefix to the volume name of the

disk in the internal disk drive. When you press this key, the Prefix line of the header will change to reflect the volume name of the disk in the internal disk drive.

If you wish to set the ProDOS date from the <NO DATE> entry in the header to the date that the transfer is being made, do so at this time by pressing the "D" key and enter the date that the files are being converted. This step ensures that you will have a record of when the conversion was made. This date will then be displayed in the MODIFIED column of the ProDOS Directory List of the volume used to store the converted files.

You may use the "R" key to reverse the direction of the file transfer if you wish to place the ProDOS disk in the external disk drive, and the DOS 3.3 disk in the internal drive for the conversion process. You may use the "C" key to record the change in slot and drive of the DOS 3.3 disk if you do reverse the disk locations. As soon as the disks are properly located as shown in the header used above and the ProDOS Prefix has been set, press the "T" key. This causes the screen below to be displayed:

```
      Transfer (or List) Files
Direction: DOS 3.3 S6,D1 ---> ProDOS
Date: <NO DATE>
Prefix: /TEST.DISK/
                     ESC: CONVERT Menu
---------------------------------------




What DOS 3.3 file(s)?

(                                     )

---------------------------------------

Press RETURN for a list of files
```

A cursor is located within the parentheses. If you know the filename of the DOS 3.3 file to be converted, enter the name and press the RETURN key. If not, press the RETURN key without entering a filename. After the RETURN key has been pressed, the catalog of the DOS 3.3 disk will be displayed on the screen. You may scroll through this catalog by pressing the right arrow key (→) to scroll down the list and the left arrow key (←) to scroll up the list. When the cursor rests on the name of the file to be converted, press the space bar.

If you change your mind, you still have a chance to cancel the command before starting the conversion process. This cancellation is accomplished by pressing the space bar one more time. Once you are sure that the indicated file is the correct file for conversion and have pressed the space bar once, press the RETURN key and the conversion process will take place.

For example, if you have followed all of the steps described above with your DOS 3.3 *System Master* disk in the external disk drive and the ProDOS formatted disk in the internal disk drive, a list of *System Master* disk files will be displayed on your screen. Scroll down this list with the right arrow key until the cursor rests on the file called COLOR DEMOSOFT. Press the space bar. A message that says "To be transferred to: COLOR.DEMOSOFT" will be printed in the bottom portion of the screen under the file list. This message indicates the new filename that will be given to the file when it is transferred to the ProDOS formatted disk. Notice that the embedded space in the DOS 3.3 filename has been replaced by a period to be consistent with ProDOS naming conventions. The use of the Prefix in the header automatically tells the program to save the file on the disk with the volume name that is the same as the prefix. Press the RETURN key to accomplish the conversion.

CONVERTING WITH ONE DISK DRIVE

If you have no external disk drive attached to your computer, the conversion process may be accomplished with only the internal disk drive. You must first insert the *ProDOS User's Disk* in the internal drive and select the CONVERT FILES option. Then, to ensure that you can make a file conversion with only the internal disk drive, you will be required to use the spare memory called /RAM as the other drive. This is accomplished first by ensuring that the direction of transfer is shown in the header as DOS 3.3 S6,D1 → PRODOS. Then you must use the "P" option to set the ProDOS Prefix.

After you press this key, you will have two choices. The first choice is a "P" which allows you to change the prefix by pathname. If you use this option, you will get a screen that asks for the desired prefix. You would enter RAM followed by RETURN. If you chose the second option by selecting "S", you would be able to designate the prefix of the destination volume by slot and drive. You would enter a "3" for the slot and a "2" for the drive. The prefix at the top of the display would change to read /RAM. Then the SYSTEM MASTER

should be placed in the internal disk drive and the "T" key should be pressed.

If you are not sure of the name of the file to be transferred, press the RETURN key. Once this causes the catalog of the SYSTEM MASTER to be displayed, scroll down the catalog with the right arrow key and press the space bar when the cursor rests on the file called MAKE.TEXT. The display at the bottom of the screen will then show that the transfer will be made to the ProDOS Prefix of /RAM under the filename MAKE.TEXT. Press the return key and the file will be converted and transferred to /RAM.

Once this has been done, press the ESC key and remove the DOS 3.3 SYSTEM MASTER from the internal disk drive. Insert the *ProDOS User's Disk* and press the "Q" key to quit. This results in a message verifying that the BASIC.SYSTEM on this volume will be loaded next if the RETURN key is pressed. Press RETURN. The *ProDOS User's Disk* Main Menu will be displayed next. When this menu appears on the screen, press the "F" key to obtain the Filer Menu. Then press the "F" key to obtain the File Command Menu. Next, press the "C" key to copy files from one disk to another. When the COPY FILES screen appears, enter /RAM/MAKE.TEXT at the first cursor position and press return.

Remove the *ProDOS User's Disk* from the internal disk drive and replace it with the ProDOS formatted disk, which will store the MAKE.TEXT file. Enter the volume name of this disk at the second cursor position. In our case, this name was /TEST.DISK, so we entered /TEST.DISK/MAKE.TEXT and pressed the RETURN key. When the copy has been made, the COPY COMPLETE message will appear on the screen. This is admittedly an awkward way to convert files, but it works for a single-disk system. If you intend to convert or transfer a large number of files from one operating system to another, be sure to have a two-disk drive system.

DISPLAY SLOT ASSIGNMENT, DISPLAY/SET TIME, AND APPLESOFT BASIC

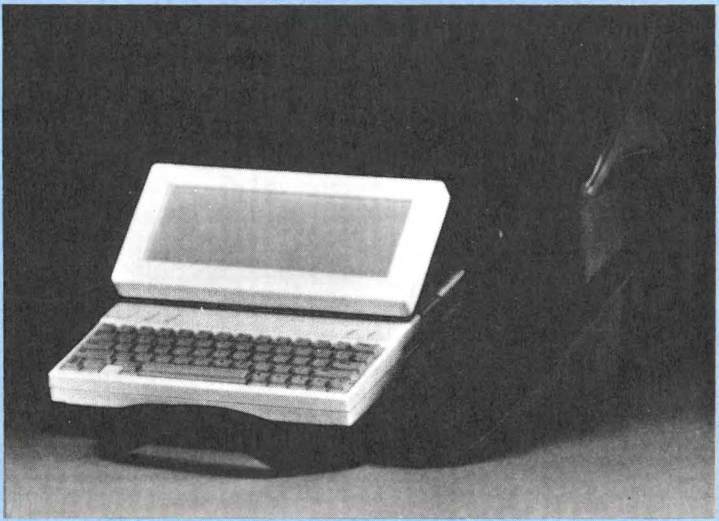
The final three options on the *ProDOS User's Disk* Main Menu are the DISPLAY SLOT ASSIGNMENT option, the DISPLAY/SET TIME option, and the APPLESOFT BASIC option. The first of these options causes the program to read each of the slots or pseudoslots, in the case of the Apple //c, and display the contents of the slots. On the Apple //c, the following display is presented:

```
****************************************
*                                      *
*        DISPLAY SLOT ASSIGNMENTS      *
*                                      *
****************************************

   STARTUP DISK: /USERS.DISK/

   Your Apple //c HAS:

   128K OF RANDOM ACCESS MEMORY

   APPLESOFT IN ROM

   SLOT 1: I/O CARD
   SLOT 2: I/O CARD
   SLOT 3: 80 COLUMN CARD
   SLOT 4: JOYSTICK
   SLOT 5: EMPTY
   SLOT 6: DISK DRIVE
   SLOT 7: EMPTY

PRESS RETURN TO DISPLAY MAIN MENU
```

The DISPLAY SLOT/TIME option allows the user to reset the time and date manually on the Apple //c. The screens are self-explanatory. The APPLESOFT BASIC option takes the user from the *ProDOS User's Disk* Menu and into Applesoft BASIC, which is stored in ROM on the Apple //c. ProDOS is still resident in memory and may be accessed from BASIC. A full range of BASIC commands are available to access ProDOS, which is stored in the RAM area of the computer's memory. These BASIC commands, the BASIC commands that deal with DOS 3.3 and the remaining commands of the BASIC language, will be discussed in Chapter 8. Among the commands to be discussed are commands that allow you to load and use programs that are stored on ProDOS initialized disks. The *ProDOS User's Disk* Menu does not have a load and run feature. It provides only a file management system, not a file use system. In the meantime, we hope that you have gotten a good grasp of how to use ProDOS from this chapter. For a list of ProDOS commands used with Applesoft, see the reference card at the back of the book. If you want to snoop more deeply into how ProDOS organizes a disk, use ProSNOOP from the *Apple //c Explorer's Disk* available in conjunction with this book.

8

Writing Your Own Programs in Applesoft BASIC



While there are thousands of programs available for your Apple //c, you may want to learn how to make the computer do exactly what you want. Since everything the Apple //c does must be directed by a program, it's required that you learn something about programming. Fortunately, programming is relatively easy to learn—much easier than learning a foreign language. Think of BASIC as a special language in which you hold a conversation with your Apple //c. At first the conversation will be rather slow, limited both by your newness with Applesoft and the computer's very limited ability to understand what you mean unless you express it in flawless BASIC. But as you progress, the dialog will become smoother and even more enjoyable, until you are able to quickly and easily give instructions to your Apple //c to do anything you might want it to do. Of course, solving very complex problems will take some time even when you are fluent in BASIC, but the rewards are proportionately greater as well.

Your Apple //c comes with a disk tutorial on BASIC called *Getting Down to BASIC*. This is an excellent introduction to BASIC, along with the material contained in Chapter 4, Programming, in the manual *Apple Presents the Apple //c*. It would be a good idea to read that chapter and go through the tutorial material on the *Getting Down to BASIC* disk as an introduction to BASIC.

In this chapter we will deal with how to think about BASIC programming and begin to explore the language a bit more. But we will also take it slow and easy and repeat some of the points from *Getting Down to BASIC* and *Apple Presents the Apple //c*. If you decide to undertake a serious study of BASIC you may eventually want to buy Apple's three books on Applesoft called *Applesoft Tutorial*, the two-volume *Applesoft Reference Manual*, and the *BASIC Programming with ProDOS* manual. You may also want to have Brady's disk, the *Apple //c Explorer's Disk*.

The *Apple //c Explorer's Disk* is a disk featuring eight useful and instructive Applesoft programs for the Apple //c, written by the authors to expand on some of the topics covered in this book. The *Apple //c Explorer's Disk* also provides such valuable services as setting up the Apple Imagewriter and Scribe printer for special printing features, examining your ProDOS disks on a sector by sector, byte by byte basis, and more. A number of additional books are listed in the appendices. In the meantime, this chapter will take you quite a bit farther into the world of Applesoft BASIC.

Mastering the BASIC language requires that you learn four fundamental concepts, which you already know and use in everyday life. You currently know and use these ideas in conversation, particularly in conversation with an unequal partner such as a small child. These concepts are: the nature of a set of instructions (computer program),

how to use general names to give instructions before the specifics are known (variables or data items), the order in which instructions are carried out, and the totally literal way computers (and young children) interpret their instructions. This chapter will teach you everything you need to know to write programs to do useful things. There are, of course, many advanced aspects of BASIC that you can go on to study with the help of a book on BASIC programming and your BASIC reference manuals. But you will find these additional features easy to learn once you have a good foundation in the fundamental concepts of programming.

This chapter is devoted to explaining four concepts, and showing in detail how to use them to write your own programs in BASIC. You will not be an expert programmer after reading these few pages. You will, however, have a way of thinking about computers and programs that relates to your everyday experiences with conversations and giving directions. You will also be aware of the differences between conversations with people and conversations with computers. And you will know the techniques needed to deal with these differences. This should get you off to a good start toward mastering BASIC through further reading and experience with programs. And most importantly, it may help you to enjoy programming in the same natural way that you enjoy the most closely related human activity—conversation with another human being—and the sharing of instructions for accomplishing the tasks that make our high standard of living and culture possible. Let's start by briefly defining each of these fundamental concepts, then go on to learn in detail how to use them.

Fundamental Concept One: Written Instructions for Accomplishing a Task (Computer Program)

You have had years of experience in giving instructions to someone else to accomplish a task. Whether they are verbal instructions to a new person at work, instructions to a teenager about yard work, directions to a repair person, or directions for finding a place in your area, all of us give directions naturally and easily as a part of our everyday life. A computer program is a sequence of instructions written in a computer language such as BASIC. It is very similar to a set of written instructions that you might provide for a new employee on a job, specifying what needs to be done, how to do it, and what to do with the results of the work once it is done. If the program is properly constructed, it will direct the computer to perform some useful task such as keeping records of accounting data, calculating numeric results, or keeping data files up to date.

If any of the instructions provided are not properly written according to the rules of the BASIC language, a SYNTAX ERROR results. The computer makes no effort to determine what you might have meant—it refers the whole matter back to you, the programmer, for correction. In addition, it is possible that each instruction you have written in a BASIC program complies to the rules of BASIC, but that the instructions do not reflect the correct steps to solve the problem you had in mind. In this case the computer will carry out the steps you wrote, rather than the steps you meant to write or "should" have written. This may result in the computer doing something similar to what you had in mind, or something radically different, or nothing useful at all, depending on exactly what steps you left out, what extra or erroneous steps you put in, or what errors of detail you may have made in preparing the program. The computer in these respects behaves very much like a young child or "simple-minded" person.

Fundamental Concept Two: The computer can not supply any missing steps or considerations. It has no common sense or experience on which to draw.

The most important point to learn here is that the computer is different in a very important way from an adult person who does a task. Any person who has grown up in our culture and attended our public schools, watched television and movies, and survived a number of years in society, has acquired a vast pool of knowledge about the way the world works and how things are done. Most of our experience in giving instructions deals with such people who have this vast store of knowledge and experience to draw on. In view of this, we tend to give very abbreviated instructions, leaving out the "obvious" details and things that "anybody would know."

The very first thing you will grasp on your way to becoming a successful computer programmer is that the situation with a computer is totally different. The computer has absolutely no experience of how the world works and how things are done. It has no way of even learning these things. All it can do is perform a very limited set of functions in exact accordance with a prepared set of instructions. If any step or detail is left out of these instructions, the computer has no store of similar experiences it can use to "figure out what you must have meant."

It will perform the instructions you give with the absolute rule that everything is explicitly stated, and only what is explicitly stated will be done. In addition, the instructions will be carried out with a totally simple-minded literalness. No modifications will be made

based on what would be "obvious" exceptions, secondary rules, or error situations to a person carrying out the same instructions. Part of the value of learning to program a computer lies in learning how much of what a person does in following a set of instructions is actually supplied by the person rather than being explicitly contained in the instructions. "You know what I mean" is never an appropriate statement to a computer—you may rest assured it does not know what you mean. It can, however, carry out such simple operations as storing names and numbers, arithmetic, and deciding which of two numbers is larger or which of two names comes first alphabetically. And it can do these extremely fast and accurately. The programmer's task is to take a problem, break it down into explicit directions, and then prepare these instructions in the format the computer can understand—BASIC statements.

In some ways writing a program for a computer is like writing instructions for a new-born child who has no knowledge of the world. Of course, new-born children do not have a built-in programming language like BASIC. Still, if you think of the Apple //c running Applesoft as a new-born requiring a very careful and detailed explanation of even the simplest tasks, you will be much more successful than those who persist in thinking of the computer as an "electronic brain."

Fundamental Concept Three: "Whoever calls" — names for information that is not specifically known yet (variables).

Giving names to information that is not totally known is a very simple concept that you use every day. Suppose, for example, you are going out and you want to arrange for someone to take phone messages for you. If the person is young, you will give fairly explicit directions such as, "If anybody calls for me, please write down their name and number so I can call them back." With an adult, we might give an abbreviated set of instructions such as, "Please take a message if anyone calls" or even just, "Would you answer the phone," assuming that they have enough "common sense" to write down the name and number of anyone who tries to reach you.

In the instruction "write down their name and number," two variables are used. "Their name," the name of any person who may call, is a variable. You want to talk about the name of a person who might call in order to give instructions to write this name down. Since you do not know who might call, and you want the name recorded no matter who it is, you use a general term to stand for the name of

anyone who calls: "their name." Then, with this variable available, you can give a very general instruction: write down their name. Notice that this cannot be accomplished by using a specific name, such as GARY PHILLIPS. If you give a child the instruction, "If Gary Phillips calls, write down his number so I can call him back," it is very likely that you will never learn the name and number of anyone else who may call. If you do, it is because the child has creatively expanded the instructions you gave to include anyone who calls. As you recall from Fundamental Concept Two, computers never expand on the instructions you give.

So, you have been using variables every day of your life, whether you knew it or not. Anytime you refer in conversation to information by a general term rather than the specific data involved, you are using variables. Let's look at some more examples. You may send a child to the store and instruct them to pay the price for a loaf of bread and bring back the change. Since you may not be sure what a loaf of bread will cost exactly today at this store, you cannot tell the child exactly how much to pay (or exactly how much change there will be). Instead, you use the general terms "the price of a loaf of bread" and "the change." These are called variables in mathematics and in computer programming. A variable is really an astoundingly simple concept that every human being uses many times every day, yet some teachers and writers have managed to make it seem mysterious. It is not. A variable is a term you use to represent something when you do not know in advance exactly what it is going to be. Using variables is an essential part of any language, including English, so that you can give general instructions that can be carried out no matter what the specific facts turn out to be. Using variables such as "the change" and "the name of anyone who calls" is fundamental to our everyday conversations and instructions to other people. Variables are equally important in writing instructions for computers. We will examine variables in more detail later.

Fundamental Concept Four: Instructions in a computer program are carried out by the computer in exactly the order in which you arrange them. However, you can give instructions that explicitly direct the computer to restart the sequence of instructions at a specific point.

When we give someone verbal instructions, it is usually understood that the instructions are to be carried out in the same order

that we give them. If we are writing a long list of instructions, it is common to number them for easy reference should there be questions or problems with any of the steps. We also find it necessary to plan for situations that may or may not arise. Let's continue with our example of sending a child to the store.

Suppose you want a loaf of French bread, but the store is often out of French bread. If we send a young child out with instructions to "buy French bread," it is possible the child will come all the way back to tell us the store is out of French bread. Very possibly we would have liked the child to buy sliced bread if the store is out of French. So, before sending the child, we give instructions to buy French bread, but if the store is out of French bread, then buy sliced bread. We have given instructions that include an appropriate response to either of two possible situations: the store does or does not have any French bread. This is something each of us does practically every day. Giving instructions that contain alternatives based on possible circumstances is as natural to adult humans as breathing. And this is all computer programming amounts to: giving instructions that include alternative actions for various situations that might arise.

If computer programming is so closely similar to things we do in everyday life, why have some people found it difficult? There are a number of reasons for this. Many people have learned to dislike mathematics and think of it as impossibly difficult because of the way it was presented to them in the public schools. Many programming books and teachers present programming as if it were a branch of mathematics, and they arouse the same feelings of dislike and incompetence that were originally instilled by the public school mathematics teachers. You will be glad to know that programming is really a branch of conversation, not mathematics. If you hate to talk to people, then you may not like programming. But if you enjoy talking to people, you will also enjoy talking to your Apple //c in the BASIC language that it understands.

This brings us to some other reasons why some people have mistakenly considered programming difficult. Computers have not yet been programmed to speak English. This means that if you want to talk to a computer, you must learn a new language such as BASIC. This is easy for most people to do, even if they are not good at foreign languages. BASIC is an extremely simple language. It has about 200 verbs to specify actions the computer can take, and three types of nouns to specify the things it can act on. This is dramatically less than the number of verbs and noun types in foreign languages. And, even better, you can get by quite nicely in BASIC once you know about a dozen verbs and the rudiments of the three noun types. The rest you can learn gradually as you write interesting and enjoyable

programs, much as you learn new English words by encountering them in your reading.

Many books and teachers of BASIC try to present the whole language, full-blown with all of its features, right away. They ask you to run before you can crawl. Worse yet, they either provide no framework to make sense of what programming is all about, or they provide a framework based on mathematics that is not congenial to most non-techincal computer owners.

In this chapter, we use as our model for creating programs for a computer, the process where you create spoken directions for other people. This is something you do every day, and we suspect it is very easy for you to do. Perhaps you even enjoy talking to people, as we do. In this case, computer programming is bound to end up on your list of favorite activities, since as you progress in programming you will soon be holding a two-way dialog with your Apple //c computer.

One final point about why some people have come to the incorrect conclusion that computer programming is difficult. Remember from the four Fundamental Principles that computers require very detailed instructions, since they have no base of knowledge and experience to draw on to fill in omitted steps. Every possible circumstance that may arise must be planned for in advance, since the computer has no common sense, and it lacks even the ability to ask you "intelligent" questions about what to do if unforeseen circumstances arise. Giving instructions to a computer is in these ways different from talking with an adult human being. An adult can fill in lots of missing steps in a set of directions, and apply "common sense" to deciding what to do in cases not explicitly covered by the instructions. This filling in of steps and handling unspecified exceptions does not always happen in exactly the way you might have liked, however. Often, what seemed like a good common sense solution to the person doing a task has disastrous consequences.

To a large extent, what we mean by "intelligent" in people is the ability to fill in missing steps in instructions, and to respond to unforeseen circumstances in ways that fit with the larger context within which the work is done. From this perspective, computers have an IQ of zero. All of the intelligence must be supplied by the programmer. The Apple //c offers only unswerving devotion and extreme speed to the simple-minded execution of your detailed instructions. It is up to you to find ways of breaking complex tasks down into simple, detailed, and above all, foolproof instructions. People who have done this have been able to use computers to help develop music, art, and literature as well as more mundane tasks such as accounting and record-keeping.

A SAMPLE BASIC PROGRAM

Now let's get our feet wet with some actual instructions to the Apple //c! For starters, let's follow the steps a programmer would take to prepare a program. Consider a simple BASIC program that instructs the Apple //c to take two numbers that you key in from the keyboard, add them together, and show the result on the screen (monitor or TV) you have attached to your Apple //c. We could write these instructions out in English first, making them more detailed and numbering the steps as follows:

1. Let me key in the first number.
2. Let me key in the last number.
3. Calculate the answer by adding the first number and the last number.
4. Show the answer on the screen.

In order to turn these simple English instructions into a BASIC program, we need to know some of the rules of BASIC. In BASIC, every instruction must begin with a number. These numbers must be in order. Our English instructions are already numbered, so we are covered there. Next, the names of variables must be in a very specific format. Have you noticed that the English instructions above use three variables? They are: the first number, the last number, and the answer. In BASIC, a variable that stands for a number must be a single name, without spaces or punctuation marks. Also, while the variable can be as long as 238 letters or numbers, only the first two count. The rest are essentially there to help you remember the significance of the term when you read the program, but they are ignored by Applesoft BASIC. Because of this little peculiarity of Applesoft BASIC, if we were to attempt to name our two variables THEFIRSTNUMBER and THELASTNUMBER, we would be in trouble. Applesoft BASIC would consider them to be exactly the same variable because it only looks at the first two letters. So let's name our three variables as follows:

English	BASIC
the first number	FIRSTNUMBER
the last number	LASTNUMBER
the answer	ANSWER

Our three BASIC variable names are different in the first two letters, are less than 238 letters and numbers long, and have no spaces or punctuation marks in them. Now we can turn to the verbs. There are four different verbs in our English instructions: key in, calculate,

add, and show. Each of these has a corresponding BASIC verb that is similar but slightly different. The corresponding BASIC verbs are:

English	BASIC
key in	INPUT
calculate	LET . . . =
add	+
show . . . on screen	PRINT

We have used three dots (. . .) wherever something else has been inserted to make a complete unit of the language, whether in English or in BASIC. Now, let's try a first, very straightforward and common sense kind of translation of our English instructions into BASIC that the Apple //c can understand:

English	BASIC
1. Key in the first number.	1 INPUT FIRSTNUMBER
2. Key in the last number.	2 INPUT LASTNUMBER
3. Calculate the answer by adding the first number and the last number.	3 LET ANSWER = FIRSTNUMBER + LASTNUMBER
4. Show the answer on the screen.	4 PRINT ANSWER

The instructions on the right in BASIC are a computer program that tell the Apple //c to do the same things specified by the English instructions on the left. Notice some of the fine points of the translation. There are no periods at the end of the instructions in BASIC. This is not allowed and would cause a syntax error. The order of words in the third instruction had to be changed around to a fixed order required by BASIC. In English we could say:

> Calculate the answer by adding the first number and the second number

or

> Calculate the answer by taking the first number and adding the second to it

or

> Add the first number to the second number to calculate the answer

or any of dozens of other ways to formulate this instruction. BASIC is much less flexible. The verb LET is optional in Applesoft BASIC, but when it is used it always comes first in any calculation instruc-

tion. (Or it could be omitted altogether.) The next thing in the instruction must be the name of the result or answer, followed by the equal sign (=). To the right of the equal sign come the variables to be used in the calculation, with the signs for the various types of calculations written between the variables in the style used in grade school math books: 5 + 6. More complex calculations can be described using parentheses exactly the way the problems were written in your grade school arithmetic books, for example: (5 + 6)/12. In BASIC, as in everyday speech, we use variables as well as specific numbers like 5 and 6. The symbols for operations in LET statements are:

English	**BASIC**
addition	+
subtraction	−
multiplication	*
division	/

To calculate Earnings, multiply regular hours by pay rate, then multiply overtime hours by 1.5 times pay rate and add the results.	LET EARNINGS = (REGULARHOURS * PAYRATE) +(OVERTIMEHOURS * 1.5 * PAYRATE)

We now have a BASIC program constructed on paper. How do we get it inside the computer so we can watch the Apple //c obey our instructions? When you have decided what program instructions you want in the computer, you must type them in to Applesoft BASIC. The best way to do this is to use a copy of your ProDOS disk that is not write-protected (the notch on the right hand side of the disk is open, not taped over). This is important because we will want to SAVE our programs on disk, which requires a disk that is not write-protected. To get to Applesoft BASIC, place a backup copy of your *System Utilities* disk or a formatted blank disk in the diskette drive, then turn the Apple //c power-on switch to the on position. Assuming everything is hooked up correctly, you will see the first menu. Option 9 is for "Exit System Utilities." This takes you to Applesoft BASIC, so select it by pressing "9" on the keyboard, which will move the highlight to <EXIT SYSTEM UTILITIES>. Press RETURN and you will be asked "Are you sure you want to leave System Utilities?" The <YES> reply is already highlighted, so just press RETURN again.

At the top of the next screen you will see a left bracket (]). This is followed by a flashing "checkerboard" cursor indicating where the character you type will appear on the screen. Notice that the Applesoft screen is a 40-column screen (with a large] and checker-

board cursor). If you have a monitor and want to work on the 80-column screen, press ESC followed by "8." This will place you in 80-column mode. You can return to 40 columns by holding ESC followed by "4." The] is Applesoft's abbreviated way of saying hello.

At this point you may begin keying in the Applesoft program that we just developed. A few additional pointers may help smooth the way. Type in the first instruction just as it appears above. Do not end it with a period. Press the RETURN key (located at the right hand edge of the computer) when you get to the end of a line. This tells Applesoft you are through with one instruction and ready to start the next. It also keeps the listing of your program on the screen neat. If you are using the 40-column screen, some of your Applesoft BASIC lines will continue from the right edge of the screen back to the left edge on the next line. Applesoft understands that this is a continuation of the same statement and handles it correctly even if a word is broken in the middle at the right edge of the screen. In fact, if you need to, you can continue an Applesoft BASIC statement for about six lines in the 40-column mode or about three lines in the 80-column mode. When you have typed all four instructions into the computer, after pressing RETURN for the end of the fourth line, type

```
LIST
```

Press the RETURN key after the T, as you must do after typing in most lines to the computer. This will cause Applesoft to list your program for you on the screen. Proofread it against the original in the book. If there are any errors, retype the one affected line correctly. Applesoft BASIC will accept this new instruction as a replacement for the old one that had the same number. If you retype line 3, the new version will replace the old version, and all the instructions will still be in their correct order even though you typed the new line 3 after line 4. Applesoft always keeps your instructions in numeric order, irrespective of what order you may type them in. There are better ways to make corrections, but they are complex enough that learning them should be a separate issue from learning the principles of BASIC programming.

Now you have your sample program inside the Apple //c's memory. To instruct Applesoft to run (to carry out or execute) your instructions, type RUN (followed by RETURN, as is almost everything you type into the computer).

Provided that everything has been entered correctly, the next thing you see will be a question mark (?). This is Applesoft giving you the chance to key in the first number. Let's use 3 to keep it simple. Press RETURN after 3 so Applesoft will know you want 3 rather than 35. If

you do not press RETURN, Applesoft has no way of knowing if 3 was all you meant to type or if you are working on 35 and you have not gotten to the 5 yet. Now Applesoft gives you still another question mark prompt (as it is called). This is to prompt you to key in the last number. Let's use 4. When you press RETURN to acknowledge that you are finished returning the last number, Applesoft will continue carrying out your instructions. The LET statement performs the addition of the two numbers you have keyed in, without any external sign. The changes it produces are inside the memory of the Apple //c. The last instruction, number 4, tells Applesoft to PRINT the result, called ANSWER, on the screen. In a flash you see 7 on the screen. You have just successfully run the first computer program written by you!

If you got a SYNTAX ERROR IN message followed by a line number, something in that line was keyed incorrectly. Rekey the offending line. If the final answer was incorrect, or you did not return two numbers and see a result, some error exists in the way the program was keyed. LIST the program, then proofread it against the original in the book. Applesoft, as all other computer languages, is infinitely picky. Even the tiniest, most trivial error in keying in a program can result in wrong results. This becomes especially evident in more complex programs.

Now that you have written an Applesoft program, it would be nice to save it so you can run it again tomorrow without having to carefully retype it. It is simple to save your program on diskette for later use. The reason we told you to start this session with a ProDOS disk that was not write-protected was so you could save your programs once they were written. To save a program, just type SAVE then the name you want to call the program, then RETURN. Program names for ProDOS start with a letter, followed by up to 14 letters, numbers, and periods. We could call this program SAMPLE.1 (and we recommend you do so to keep things simple). Try it!

Unless something went wrong, you heard the disk turn and saw Applesoft put its friendly] back on the screen, as if to say "What next, partner?" If you saw the message, SYNTAX ERROR, type this statement again:

```
SAVE SAMPLE.1
```

Remember to press RETURN at the end of the statement. Now you have a copy of the program stored on the diskette under the name SAMPLE.1. You can see some evidence that the program really is SAVEd on the diskette by typing CAT. This is a request to ProDOS to show you a list of the names of the files (both data files and programs) recorded on the diskette. There is a lot of other infor-

mation on the resulting display that we are not concerned with now. Just verify that the name SAMPLE.1 appears in the list of file names on the diskette. If it does not, something went wrong in your effort to save the file, and you may try it again. SAVEing a program ordinarily works just fine. If it does not, there is a specific reason. Perhaps you miskeyed the SAVE instruction, a write-protect tab was on the diskette, or some other condition prevented successful saving of the program.

Now let's try a little exercise that will both convince you that your program really is permanently stored on the diskette, and teach you more about using BASIC and ProDOS. Let's assume that you got a successful save of your program to diskette, evidenced by Applesoft coming back with a] and no error message after your SAVE command. Now type LIST. (We think it is probably time to stop reminding you to press RETURN at the end of each line you type into your Apple //c, don't you?) You will see a list of your four instructions to the computer. These instructions were LISTed from the copy of the program that is inside the memory of the Apple //c. Another copy is recorded on the diskette as SAMPLE.1.

Since we have an extra copy on the diskette, let's eliminate the copy now in memory. To do this, type NEW. This instruction is intended to let you eliminate anything now inside the Apple //c's Applesoft program memory and start a new program from scratch, but will not affect the diskette. The copy of the program stored on the diskette is still safe and sound. To demonstrate that the NEW command eliminated the program, type LIST again. Nothing is LISTed, indicating that there is no program present. To further illustrate this, type RUN. Again, nothing happens because there are no instructions currently recognized inside the Apple //c's memory. There were instructions there earlier, but we eliminated them with the NEW command.

Now, let's load our program back into memory from the diskette. To do this, type LOAD followed by the name of the program (SAMPLE.1). The statement will be

```
LOAD SAMPLE.1
```

Applesoft will reply with] if all goes well. If you get a SYNTAX error, try again and type more carefully. If you get "PATH NOT FOUND", this is the way ProDOS lets you know it could not find the program under the name you gave. There are at least three possible reasons for this:

1. When you originally saved the program, you mistyped the name. The program is stored under a name other than SAMPLE.1, perhaps SAMPLE1 or SAMPLE, or whatever. To find

out the actual name the program is saved under, type CAT. This asks ProDOS to show you a list of all the files on the diskette (from CATalog, as in Sears-Roebuck). From the resulting list of files (some data, some programs with TYPE or BAS for BASIC), you should be able to see what name the program is saved under. If there is nothing on the CAT listing similar to SAMPLE.1, then see step 2.

2. The program was somehow not saved. You should have gotten an error message in this case. If you are sure you did not get an error message after "SAVE SAMPLE.1", you may have to start the whole process over from turning on the computer. Something went wrong, and the number of possibilites is too great to justify more than about three seconds of wondering what it was. With greater computer experience, you will have a better handle on how things are going and what, if anything, may have gone wrong.
3. The program is saved correctly, but you mistyped the LOAD command with a spelling for the program name different from SAMPLE.1. Try again.

Once you have a successful LOAD (hopefully it was the very first time you tried it), you can again use LIST to verify that you have in fact successfully written, RUN, SAVEd, and LOADed a BASIC program. You are well on your way to fluency in BASIC, and it is a lot simpler and more fun than a foreign language!

SUMMARY OF THE FIRST SAMPLE

In the course of working out this first sample program to add two numbers and PRINT the result, we covered quite a few things. Perhaps it would be wise to stop and review these before going on to your second BASIC program.

You have learned:

1. How to type in BASIC statements and retype lines with errors.
2. What SYNTAX ERROR means and how to correct it.
3. How to name variables for Applesoft BASIC programs (they start with a letter, only the first two characters count, they are up to 238 characters long with letters and numbers, no spaces or punctuation marks, and no reserved words inside variable names).
4. How to translate English instructions into BASIC instructions.
5. How to name a program in ProDOS (it must start with a letter, then up to 14 more letters, numbers, or periods).

6. How to get a list of all the files (data files and programs) on a diskette (CAT).
7. How to instruct Applesoft to perform a calculation (LET . . . =).
8. How to instruct Applesoft to do arithmetic (+, −, *, /).
9. How to instruct Applesoft to allow you to key in a number and save it in a variable name (INPUT).
10. How to instruct Applesoft to show a result on the screen (PRINT).

Ten new skills is a pretty good helping for one lesson, so if you are feeling like a break, this is a good point to stop and let this all soak in a bit. Of course, if you are eager to keep going, be our guest!

A FEW DETAILS WE GLOSSED OVER

In constructing the first sample program, we deliberately kept things as simple as possible. The point was to convey the most fundamental issues about writing a BASIC program without cluttering up your attention with details. But BASIC is much richer than the first sample program may have lead you to think. As a second sample program, we will take the first and jazz it up a bit by including some of the optional features we omitted before.

We can also take time to explain a number of points that we did not mention before in an effort to keep things as simple as possible. First, the program listings are all in upper case letters. This reflects Applesoft's tendency to put all of the special BASIC words such as PRINT and LET into capitals even if you key them in lower case. It really does not matter which way you key them. These BASIC special words are called reserved words, and you will need to understand a little bit about how Applesoft treats them when you start to write your own programs. A reserved word may not be used in a variable name, even in the middle of the name. If we had chosen the name SECONDNUMBER for the program above, we would have been in trouble. ON is an Applesoft reserved word. If you key in the statement

```
10 INPUT SECONDNUMBER
```

Applesoft will change it to

```
10 INPUT SEC ON DNUMBER
```

and give you a SYNTAX ERROR IN 10 message when you try to run your program. There are so many reserved words in Applesoft BASIC it is hard to avoid ever using a variable name that has a

reserved word embedded in it. The main point is to recognize what is causing this transformation of the statements you have typed in and fix them. This requires finding a name that will still serve to remind you of the meaning and purpose of your variable, but it does not contain any reserved words. A list of Applesoft and ProDOS reserved words is at the end of this chapter.

We have also been using variable names for numbers without explaining the three types of variables used in Applesoft BASIC. The type you ordinarily get is a number, potentially including a decimal fraction such as 23.45. There is another type of variable for numbers, called integer. This is for whole numbers ("integers") only, and is indicated by placing a percent sign (%) after the variable name: WHOLENUMBER%. Integer variables are very good for counting things or other operations that require only whole numbers because they can be processed faster than the ordinary numeric variables. The third type of variable is for storing letters and symbols as well as numbers. Called string variables, these are indicated by a dollar sign at the end: NAME$. We will discuss string variables later.

Some of the commands we have covered here are not part of Applesoft. They are ProDOS commands interpreted by BASIC.SYSTEM. These include LOAD, SAVE, and CAT. If you want to see a list of the various ProDOS commands available with Applesoft BASIC, see the list "Command Table" in Chapter 6. These Applesoft ProDOS commands provide a way of communicating with ProDOS from Applesoft BASIC.

A SECOND PROGRAM THAT IS A LITTLE FANCIER

Let's get back to our program to add two numbers. Start with the INPUT statements. It would be useful to have the program remind you of what you are supposed to key in. When you have written dozens of programs you will not always remember exactly what each is expecting you to key in. This is done by including a literal prompt, in quotation marks, in the INPUT statement. This is separated from the name of the variable by a semicolon (;). One way to improve the INPUT statements might be:

```
1 INPUT "KEY FIRST NUMBER";FIRSTNUMBER
2 INPUT "KEY LAST NUMBER";LASTNUMBER
```

We can also provide some identification of the result, rather than just putting a raw 7 on the screen. Here is one way to do this:

```
4 PRINT "THE ANSWER IS:";ANSWER
```

Even fancier would be a complete statement as the result:

```
4 PRINT "THE SUM OF ";FIRSTNUMBER;" AND
  ";LASTNUMBER;" IS
  ";ANSWER:"."
```

If you return these lines correctly along with our earlier line 3, you should get the following results when you RUN the program:

```
KEY FIRST NUMBER2
KEY LAST NUMBER3
THE SUM OF 2 AND 3 IS 5.
```

Note that we carefully allowed a space in the literal inside quotes on each side of the numbers. Had we not done this, the words and numbers would have run together. Always remember that the computer will never attend to even the most trivial detail for you. If you do not specify it, it does not happen, no matter how obvious it would be to a person doing the same job.

You may want to save this program as SAMPLE.2 for further practice in SAVEing and LOADing programs. After SAVEing SAMPLE.2, use CAT to get a list of files on the diskette and verify that SAMPLE.2 is now among them.

Now let's get some hardcopy, if you have a printer attached to your Apple //c. To do this, just add PRINT CHR$(4);"PR#1" to the PRINT statement like this:

```
4 PRINT CHR$(4);"PR#1":PRINT "THE SUM OF
  ";FIRSTNUMBER;" AND
  ";LASTNUMBER;" IS ";ANSWER;"."
```

PRINT CHR$(4);"PR#1" notifies the operating system that the PRINTed data should go to the printer attached to your Apple //c instead of to the screen. PRINTed data goes to the screen by default. To print on the printer requires that you use PRINT CHR$(4);"PR#1" once. All subsequent PRINT statements will print on the printer rather than the screen. To return to the normal situation, where everything is displayed on the screen, type PR#0.

PRINT CHR$(4);"PR#1" is actually a BASIC statement and could have been on a line by itself. To function correctly with ProDOS it should always be used as a ProDOS command in a program by using it in the full statement, PRINT CHR$(4);"PR#1". You can put several BASIC statements on the same line with a single line number by separating them with colons (:). This brings up an interesting point about line numbers. Most BASIC programmers number their

instructions by tens (10, 20 , and so forth) or hundreds, rather than by ones as we did. This makes it much easier to insert new lines in a program if you discover that they are needed later. In this example, we would have to renumber the program in order to insert the new PRINT CHR$(4);"PR#1" instruction ahead of the PRINT instruction. We sidestepped the problem by squeezing the PRINT CHR$(4);"PR#1" statement onto the same line with the PRINT statement. As a rule it is better to plan ahead and number the lines of a program by tens.

Let's move on to some new material and consider a substantially different problem. Suppose the problem is to total a list of numbers (not knowing in advance how many there are) and print the sum, the number of numbers entered, and the average. This will require some BASIC instructions we have not used so far, and some new wrinkles on old ones.

While the instructions we have just specified might be fine for a person, Applesoft will not know what we mean by "all the numbers on the list" or any such phrase. To solve this problem, programmers have invented the device of using one special input value to mean "there are no more input values–it is time to perform the calculations and print the answers." We could use zero as this special value, but it might occur in some lists and need to be counted as one of the data items for the purpose of determining how many there are and what the average is. Let's suppose that for this example there are to be no negative numbers in the lists to be added. Then we can use -1 for the "end of list" indicator. We will need to have a decision in the program: is this a data item to be counted and added with the others, or is this the "end of list indicator?" This will require the IF statement. IF is the main statement used to instruct the computer to make a decision based on the comparison of two or more data items, and to do things differently depending on how the comparison comes out. One way of restating this problem so that the instructions are completely detailed and suitable for execution by a person with no background knowledge or by a computer would be:

1. Key in a number.
2. If the number is -1, then skip to step 6.
3. Add the number to the total so far.
4. Add one to the count of data items keyed in.
5. Skip back to step 1.
6. Calculate the average by dividing the count of data items keyed in into the total so far.
7. Show the count of data items, the total, and the average on the screen.

These English instructions can be translated into BASIC very easily. Can you see that this is one way of instructing someone, in detail, to add up a group of numbers until they encounter a −1, then show how many items there were, the sum, and the average? The program would look like this in BASIC:

```
1 INPUT NUMBER
2 IF NUMBER = -1 THEN GOTO 6
3 LET TOTAL = TOTAL + NUMBER
4 LET COUNT% = COUNT% + 1
5 GOTO 1
6 LET AVERAGE = TOTAL / COUNT%
7 PRINT "THERE WERE ";COUNT%;" ITEMS TOTALING
  ";TOTAL;" AND AVERAGING ";AVERAGE;"."
```

Lines 1, 6, and 7 should be self-explanatory from our earlier programs. The others deserve an explanation. Some are brand new and others involve new aspects of the same statements used before.

Line 2 instructs the computer to compare NUMBER (whatever number was most recently keyed in at statement 1 with −1. If they are not equal, then the program continues its routine of adding and counting the numbers as they are keyed in. But if they are equal, that is, if the number keyed in for NUMBER at statement 1 was −1, the THEN clause directs Applesoft to make an exception to its normal procedure. While Applesoft ordinarily executes the instructions in a program in the strict sequence they were entered, this instruction says to take instruction number 6 as the next instruction (GOTO 6). The GOTO instruction allows us to direct the computer to make exceptions to the rule that instructions are executed in strict numerical order, and specify that the sequential execution of instructions is to resume at the specified instruction. Combined with the IF . . . THEN statement, the GOTO statement is the primary way that programs can respond differently to different circumstances.

Line 3, LET TOTAL = TOTAL + NUMBER, is another example of a calculation instruction, but with a new twist. The answer, TOTAL, is also used in the calculation. This is a way of accumulating a total of a number of items without having to have them all available at the same time. As each NUMBER is keyed in, it is accumulated into a running total of numbers keyed in so far. Suppose the total so far is 52. The value of TOTAL is 52. Now we key in the number 3 (NUMBER is 3). At statement 3, we calculate TOTAL + NUMBER, or 52 + 3. The result, 55, is placed into TOTAL as its new value, replacing the older value of 53. This is a standard programming technique.

Be of good cheer, for there are not too many of these "techniques of programming" that you will have to learn. You have learned two so far (using an end-of-list marker value and forming a running total with the SUM = SUM + NEWITEM technique). There are about a half dozen techniques you need to be a good programmer—a very small number compared to the rules of grammar for a foreign language. A master programmer, like the native speaker of a language, may have dozens more to aid in efficiency, style, and for very special situations. But to be a competent programmer you only need about four or so more programming techniques, plus a bigger vocabulary of BASIC verbs and more experience in finding errors and correcting them.

Line 4, LET COUNT% = COUNT% + 1, is the programmer's standard way of counting. It is exactly like the use of a running total in line 3, except that the fixed value 1 is being added to the count-so-far each time. This type of a running subtotal simply counts the number of times the LET instruction has been executed. In this program this number will be the same as the number of NUMBERs entered. Notice that we have used % at the end of COUNT% to make it an integer variable. A count of the number of NUMBERs entered must be a whole number, and using the % makes the computation faster. It is enough of a difference in speed to be noticeable when you write more complex programs. Still, we will not use % in the rest of the sample programs because it tends to make them harder to read and follow.

Line 5 shows us another use of GOTO. Here no decision is being made. Instead, we are making a circle through the instructions, or a loop. Rather than having one INPUT instruction for each NUMBER (we do not know how many there might be), we reuse the statements we have. This "loop" structure, where program lines are reused as many times as needed until some specific circumstance signals that we can stop going around the loop (in this case a −1 as the NUMBER value), is another standard programming technique. Now you know three, and the most important and frequently used three at that.

Loops bring with them some of the most frustrating and amusing of the problems of testing (or "debugging") computer programs. Loops have the potential to become infinite (lasting forever) if the conditions for terminating them are omitted or misstated. And when complex loops-within-loops are written in large programs, it can become difficult to find out what is going wrong. If you continue in your trek as a programmer, you are certain to encounter these sorts of problems. If you have studied this sample program carefully and really understand exactly how it works, you are now in a position to write a large number of interesting programs. You can increase your

range of possibilites even more by going through, systematically, some of the other options available on statements we have already studied.

For example, take IF ... THEN. This statement can do a lot more than we have seen. We have only used = as the condition to be tested. We could also have used any of these comparative operators:

=	equal to
<>	not equal to
>	is greater than
<	is less than
>=	greater than or equal to
<=	less than or equal to

For very complex logical tests, it is possible to group simple comparisons with parentheses and use logical connectives between them (AND, OR). This is done in exactly the same way as in everyday speech: "If you get to the beach before 3:00 and I'm not there or you run into George, just go to the lounge and wait for me there." Complex logical conditions, using AND as well as OR and an alteration in the sequence of instructions, based on a possible contingency are shown here. Beach programming!

The IF ... THEN instruction can also take instructions other than GOTO after THEN. As many instructions as you like may be strung together, just separate them by colons. It is sometimes simpler to place some instructions right in the IF ... THEN statement, that determines whether or not they are to be executed, than to place them somewhere else in the program and GOTO the line number where they are located. It is acceptable to place an IF ... THEN statement after the THEN of another IF ... THEN statement, creating "nested IFs." While the logic can get quite complex, it may be useful in the solution of some very complex problems.

One final note on the ins and outs of the IF ... THEN statement. If the THEN is to be followed by a GOTO only, you can omit the word GOTO and just put the line number. Many programmers do this consistently, so it is something to keep in mind when reading other people's programs.

Many of the other BASIC statements have a multitude of additional features, and there is no reason (or space) to go through all of them. They will make more sense if you learn about them gradually as you tackle various problems that require them, or as part of a full length book or course on BASIC. But there are a number of additional points that we would like to cover in this chapter. Some of these are so fundamental that you will need a knowledge of them for

most programs, while others are often poorly explained in manuals, books, and courses on BASIC programming.

FOR . . . NEXT—A SHORTCUT TO BUILDING LOOPS

Loops are such an important and powerful part of BASIC programming that a special pair of statements is available to make writing loops easy. These are the FOR and NEXT statements. They form a pair of brackets around statements in a program that are to be executed more than once. The FOR statement also makes it very easy to specify how many times the statements between FOR and NEXT should be executed. FOR also allows for varying the results by stepping a variable through a series of values. For example:

```
10 FOR I = 1 TO 10
20 PRINT I
30 NEXT I
```

This short program will print out the numbers from 1 to 10. At first, I is set to the value 1 because of the specification FOR I = 1. Next, all statements between the FOR and NEXT are executed. When the NEXT is encountered, it acts as an IF . . . THEN statement that returns the sequence of instruction execution to the FOR statement until the loop variable, I, reaches the limit (10) stated in the FOR statement. When the limit is reached, the program continues to the instruction following the NEXT (if there is one). Good programming practice requires you to use I% instead of I, since this is much faster. To keep things as simple as possible, we are omitting this step. The little program above is equivalent to this one:

```
9  I = 0
10 I = I + 1
20 PRINT I
30 IF I < 10 THEN GOTO 10
```

FOR . . . NEXT even allows you to step through a series of values by a step size other than 1. For example, to print the even numbers only, specify a STEP of 2:

```
10 FOR I = 2 TO 10 STEP 2
20 PRINT I
30 NEXT I
```

In order to realize the full power of the FOR . . . NEXT loop, we will need to use it with another feature of Applesoft BASIC known as

subscripted variables. This is a very simple and powerful way of dealing with data that looks like a list or table.

TABLES OF DATA, ARRAYS, AND THE DIM STATEMENT

We have already seen that there are some catches in naming data variables, since only the first two characters are considered by BASIC. When there are a large number of different data items to be used in a program, having a totally separate name for each one becomes unacceptably clumsy. Fortunately, BASIC offers us a simplified way of naming a large group of related data items, such as a list or table of numbers.

DIM is a statement that gives BASIC information about how data will be named. It causes space to be reserved, but does not perform any operations on the data. In order to understand the function of DIM, you must first understand the fundamental concept of arrays. An array is a group of data quantities, all of the same type, that forms a group to be processed in a similar way. Suppose, for example, that you need to write a program that accepts 20 numbers from the operator, then performs calculations based on all 20 numbers. A brute force approach to this problem would produce 20 names, all satisfying the rules for variable names in BASIC and all different in the first two characters: N1NUMBER1, N2NUMBER2, . . ., N9NUMBER9, N0NUMBER10, M1NUMBER11, . . ., M9NUMBER19, M0NUMBER20. The program to read these from the operator would look like this:

```
110 INPUT "FIRST NUMBER";N1NUMBER1
120 INPUT "SECOND NUMBER";N2NUMBER2
 .
 .
 .
290 INPUT "NINETEENTH NUMBER";M9NUMBER19
300 INPUT "TWENTIETH NUMBER";M0NUMBER20
```

Subsequent calculations on the numbers would be even more cumbersome. It is easy to imagine the statements required to calculate the sum, average, and other statistical measures on these numbers. Of course, this approach totally misses the point of the computer's ability to use loops. Arrays give us the ability to name the individual numbers (or character strings) in a related collection in a way that simplifies repetitive operations. To name our group of 20 numbers, we would use this DIM statement:

```
10 DIM N(20)
```

This informs BASIC that we will be using a group of 20 numbers. All of the numbers have the generic name N qualified by a number from 1 to 20 in parentheses following. The first number is named N(1). The second number is named N(2). The twentieth number is, of course, named N(20). When we combine this flexible naming convention with the ability to use a variable inside the parentheses of the data name, many kinds of processing become very easy compared to the brute force method of assigning a different name to each data item. To read in our 20 numbers, for example, we could use the following short program, which uses our old friend, the FOR ... NEXT statement:

```
10 DIM N(20)
20 FOR SUBSCRIPT = 1 TO 20
30 INPUT "NUMBER(";SUBSCRIPT;") =";N(SUBSCRIPT)
40 NEXT SUBSCRIPT
```

Notice how we broke into the prompt in statement 30 to insert the subscript (or positioning number within the group of related items) in the prompt given to the person who operates the program. Once we have these numbers in the computer, performing various operations on each member of the group is a snap. The FOR ... NEXT statement lends itself very easily and naturally to computations with subscripted variables. The following code will compute and print out the sum of the 20 numbers read above:

```
50 FOR SUBSCRIPT = 1 TO 20
60 LET SUM = SUM + N(SUBSCRIPT)
70 NEXT SUBSCRIPT
80 PRINT SUM
```

Once you get the general idea of subscripts, there are thousands of ways of using them for both numbers and character strings. Most books on BASIC programming will give many more examples of how to use subscripts with dimensioned arrays to solve common types of programming problems. Notice that we have used only a single letter in naming an array. This is a special restriction for names of dimensioned arrays.

Often data is in the form of a table, where each entry is in a certain row and column location. Data in this form requires two subscripts for unique specification, such as T(2,14). This specifies the particular number that is at row 2, column 14 of a table of related data numbers known collectively as T. Here again you must use the DIM (short for dimension) statement to let BASIC know how much space to reserve for this group of data items. BASIC also uses the maxi-

mum subscripts, stated in your DIM statement, to determine if your loops may have gone awry and asked for a data item with a subscript larger than those you planned for and have data values for. If your table of data items contains 30 columns and 20 rows, the DIM statement for it could be:

```
10 DIM T(30,20)
```

Looking at the DIM statement more generally, up to 88 different maximum subscripts may be specified for a single array name such as TABLE. While problems requiring more than two subscripts are rare, they do occur. Also, you can list as many variable names with their maximum subscript values as you like in one DIM statement, separating them with commas. The maximum subscript value is 32767, large enough for most everyday problems.

Now for some little twists. Probably because old-line computer programmers like to start numbering things from zero instead of one (to squeeze the most possible different numbers out of a fixed number of bits), subscripts start from zero instead of from one. Fortunately, you can just ignore the extra variable with subscript zero in most programs.

Next, BASIC will let you get away with using an array without a DIM statement. In this case the maximum subscript is 10 by default. This may help you whip up a quick and dirty program in a hurry. But more often than not failing to explicitly dimension your arrays and tables will come back to byte you in the end.

Some other interesting tidbits about DIM and the arrays it creates are:

- All numeric data items in arrays have a starting value of zero.
- All strings in arrays have a starting value of null—the string of length zero.
- The rules for variable types apply to arrays just as to individual variables (default is numeric, $ for strings, % for integer).

Arrays and the DIM statement are the key to short programs which process large amounts of data.

ENOUGH OF NUMBERS! LETS HAVE SOME NAMES!

So far we have been focusing exclusively on numbers. But your Apple //c and Applesoft can also do lots of things with names, addresses, and other items of information that contain letters and

other symbols as well as numbers. Data items that contain letters or symbols are called character strings or just strings for short. A string must be stored in a string variable. These are named exactly like numeric variables, but string variables must end with a dollar sign ($). Examples would be NAME$, ADDRESS$, and A$(5) for a string in a string array.

Data for strings may be keyed in with the INPUT statement in the same way as numbers. In your BASIC programs, strings are handled differently. Instead of arithmetic operations used on numbers, strings are processed with a group of operators known as string manipulations. The principle string manipulations are assigning a value to a string, joining two shorter strings together to make a longer string, and picking part of a string out to make a new string. Strings can be compared to one another with the IF statement in much the same way as numbers. In this case greater than means "comes later in alphabetical order than."

As an illustration of how a BASIC program can process strings and how the various string functions work, let's take a BASIC program apart and see how it does its work. The following program will take a name in the form of "last,first" and separate the two parts to be printed as "first" followed by "last" on the next line. The data will be entered surrounded by actual double quotes, so a sample input would be:

```
"PHILLIPS,GARY"
```

This is a fairly typical application for string processing, showing how the available string manipulation commands can be combined to perform a useful task. Names are frequently stored in computers in "last,first" format so they can easily be sorted alphabetically by last name. This program allows names stored in this way to be reassembled into the traditional "first,last" order for printing checks, addressing letters, and so forth. Of course, the real reason it is here is so we can study the string manipulation features of BASIC.

```
10 INPUT "LAST,FIRST OR END";NAME$
20 IF NAME$ = "END" THEN GOTO 110
30 FOR I = 1 TO LEN(NAME$)
40 IF MID$(NAME$,I,1) = "," THEN GOTO 80
50 NEXT I
60 PRINT "NO COMMA FOUND IN INPUT"
70 GOTO 10
80 PRINT RIGHT$(NAME$,LEN(NAME$) - I)
90 PRINT LEFT$(NAME$,I-1)
100 GOTO 10
110 PRINT "GOOD BYE!"
```

Let's learn these new string features by taking this program apart line by line and seeing how it works.

Line 10. This a standard INPUT statement with a prompt to remind the person running the program how the input should look. The double quotes are required because Applesoft usually considers a comma to be the signal that one string has ended and another is beginning. In order to have a comma in the middle of a string, the entire string must be contained in quotes. If you forget the quotes when you run this program (or try leaving them out as a learning experiment), you will get a message from BASIC: EXTRA IGNORED. This means that BASIC found two strings, separated by a comma, when only one is requested by the INPUT statement. The "extra" string is ignored. Of course, the real problem is that the entire string should have been contained in quotes, so that it was taken by BASIC as one string rather than two strings separated by a comma.

Line 20. Here we use the IF . . .THEN statement to check for a special string (END), which we are using as an "end of list" marker, much as we used the number −1 to mark the end of a list of numbers before. If the program finds END to be the INPUT, then it will GOTO line 110 and say GOOD BYE. If the INPUT was not END, but rather a name, the program continues to line 30 according to its standard rule of executing instructions in numeric sequence.

Line 30. Here we use the FOR . . . NEXT construction. The only new wrinkle here is that the progress of I is to stop at LEN(NAME$). LEN is an Applesoft built-in string function that calculates the number of characters in a string. If we run this program and key in "PHILLIPS,GARY" as the value for NAME$, then LEN(NAME$) will be 13. Even the comma (and any spaces inside the quotes) will be counted by LEN. The nice thing about this is that it allows us to go through this FOR . . . NEXT loop exactly as many times as there are characters in NAME$—once for each character.

Line 40. This IF statement does a very simple job, although it looks a little unwieldy. Let's take it apart piece by piece. MID$ is a string function that extracts a small string from the middle of a larger string. The small string will be produced according to the three items listed in parentheses after MID$. The first must be the large string from which the small string is to be taken. Here we are clearly taking part of NAME$ to produce the new, shorter string. The part taken from NAME$ will start at the character number listed in the second position after MID$, and be as long as the number in the third position (1 in this case).

The second item in the list is I. The first time through this FOR . . . NEXT loop, I will be one. MID$(NAME$,I,1) will therefore be the same as MID$(NAME$,1,1). This will pull out just the first character for comparison to a comma. If NAME$ = "PHILLIPS,GARY", and I

= 1, then MID$(NAME$,I,1) is G. Right? And when I = 2, MID$(NAME$,I,1) is A. And so forth. As we go through this loop, we pull each of the individual characters out of NAME$ one at a time so we can check each to see if it is a comma.

When the comma is pulled out by MID$, the THEN clause of the IF statement is activated, and we go to line 80 to print out the two separate parts of the name. The value of I will serve to save the information as to where in the string NAME$ we found the comma. It may be helpful to you to use a pencil and paper to draw out in detail how the string is separated out by MID$ each time through this FOR . . . NEXT loop.

Line 50. This serves to end the FOR loop started at line 30.

Line 60. If there were a comma anywhere in the input string NAME$, it would have been found by statement 40 and the program execution would be sent to line 80. The fact that we got here (line 60) means that we examined every character in NAME$ and not one of them was a comma. Since we cannot split a name into two parts if it does not have a comma between the parts, we print an error message, "NO COMMA FOUND IN INPUT".

Line 70. After printing the error message, it is polite to go back and give the operator a chance to correct the error (by retyping the name).

Line 80. The only way to get here is by finding a comma in the input in line 40 and taking the GOTO 80. At this point we know there is a comma at character position I in NAME$. That is, if I is 5, then the comma is in the fifth position of NAME$. Everything to the right of this comma is the person's first name. Everything to the left is the last name. In this statement we will print the first name only. The RIGHT$ string operation creates a small string from a large string, much like MID$. The difference is that RIGHT$ takes the rightmost part of the string, to create a string of the length specified in the second item after RIGHT$. In this case, RIGHT$(NAME$, LEN(NAME$) − I), the number of characters retained in the rightmost part of the string is LEN(NAME$) − I. Since I is where the comma is located, and LEN(NAME$) is the length of the entire name, LEN(NAME$) − I is the number of characters in the part of the name following the comma. This is the first name, so the net effect is to pick up the first name only and print it. If you are not quite sure about this, take a pencil and paper and draw out a sample name. Do all of the calculations, and you will see that this does, in fact, pick up the first name from the rightmost part of the NAME$ string.

Line 90. In a manner very similar to line 80, this PRINT statement will print out the last name only from the leftmost (or first) part of the string NAME$. LEFT$ is a string operator that produces a small string from the leftmost part of a larger string. The resulting string

always starts at the leftmost character (the first character) of the larger string. The new string will have the number of characters specified by the second item listed after LEFT$, in this case I − 1. Since I is the position in NAME$ where the comma was found, there were I − 1 characters in the last name (we have to exclude the comma itself, hence the −1). The net effect of this statement is to print out the last name only from NAME$.

Line 100. After separating one name into first and last parts, we go back to process another name (or perhaps to get the END code).

Line 110. Here we say goodbye to whoever is running the program because they entered END as the NAME$ in statement 10, and statement 20 detected this and took the GOTO 110.

While this program does take some effort to follow, it will give you some foundation in the whys and hows of string processing. It uses four of the main string operations (LEN, MID$, LEFT$, and RIGHT$), and shows the general technique of searching through a string one character at a time for a special marker such as the comma in our name.

There is a great deal more to BASIC. We hope you have come to appreciate the power and flexibility of the language. Perhaps programming in BASIC is not your cup of tea. Or, perhaps you may share some of the excitement and enjoyment that we feel when creating a set of instructions that can make the Apple //c do new and interesting things. The quick reference card at the back of this book lists all Applesoft commands and ProDOS commands used from Applesoft. If you want to go on in your study of programming, pick up the three Applesoft manuals from your Apple dealer and see the book list in the appendices for a number of other books on BASIC. The disk, *Apple //c Explorer's Disk*, available from your bookstore or directly from Brady, contains eight additional Applesoft programs to help you explore Applesoft further as well as perform valuable functions such as configuring your Imagewriter or Scribe printer for special printing or snooping through your ProDOS disks on a sector by sector, byte by byte basis.

LIST OF BASIC RESERVED WORDS

Reserved words have particular meanings in BASIC and are used for commands, statements, functions, and operator names. Variable names cannot be or contain a reserved word. This can cause some confusion since many arbitrarily chosen variable

names may accidentally contain reserved words such as AT, ON, OR, TO, and others.

Following are the reserved words in Applesoft BASIC and ProDOS:

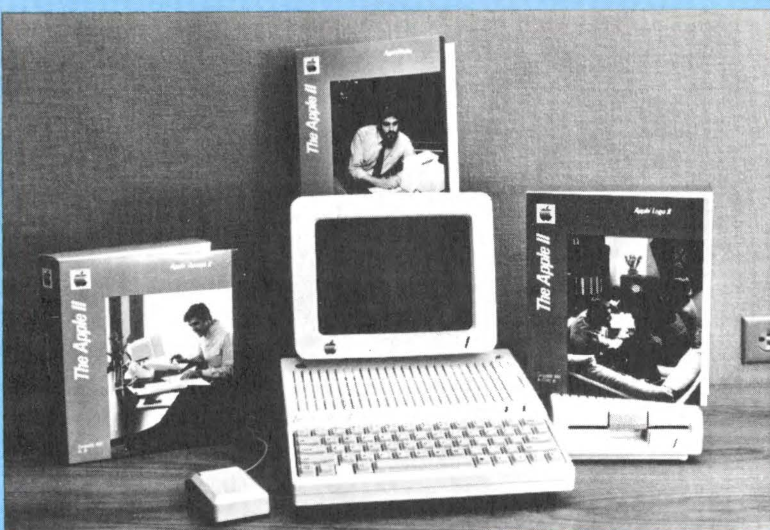
&	EXP	LOG	RND
-	FLASH	LOMEM	ROT=
ABS	FLUSH	MID$	RUN
AND	FN	NEW	SAVE
APPEND	FOR	NEXT	SCALE=
ASC	FP	NORMAL	SCRN(
AT	FRE	NOT	SIN
ATN	GET	NOTRACE	SPC(
BLOAD	GOSUB	ON	SPEED=
BRUN	GOTO	ONERR	SQR
BSAVE	GR	OPEN	STEP
CALL	HCOLOR=	OR	STOP
CAT	HGR	PEL	STORE
CATALOG	HGR2	PEEK	STR$
CHAIN	HIMEM	PLOT	TAB(
CHR$	HLIN	POKE	TAN
CLEAR	HOME	POP	TEXT
CLOSE	HPLOT	POS	THEN
COLOR=	HTAB	POSITION	TO
CONT	IF	PREFIX	TRACE
COS	IN#	PRINT	UNLOCK
CREATE	INPUT	PR#	USR
DATA	INT	READ	VAL
DEF	INVERSE	RECALL	VLIN
DEL	LEFT$	REM	VTAB
DELETE	LEN	RENAME	WAIT
DIM	LET	RESTORE	WRITE
DRAW	LIST	RESUME	XPLOT
END	LOAD	RETURN	XDRAW
EXEC	LOCK	RIGHT$	

II

Apple //c Software Buyer's Guide

9

How to Select Software for Your Apple //c



SOFTWARE—THE MAGIC INGREDIENT

Your Apple //c is a powerful computer capable of performing an amazing variety of tasks. Yet without proper software, the Apple //c will do very little (it does have built-in firmware). Finding the right software is the most important part of setting up a smooth running and useful Apple //c system.

The major purpose of this section is to provide information about the various types of software available. This chapter provides an introduction to many types of software. We will define programs such as spreadsheets, database managers, word processors, and others. It will also explain how you can use these programs.

The information contained in this chapter will aid you in selecting a program to fill a specific need. There are many programs that are widely useful tools for Apple //c users. Most of these were originally developed for the Apple II, II+, or //e, but will work fine on the //c.

There are two major areas of software for the Apple //c: commercial software and public domain software. This part of the book focuses primarily on commercial software, but we do cover public domain software very briefly.

Selecting software is one of your most important undertakings as an Apple //c owner. Your computer can do marvelous things with good software. Without good software it is very limited. Three fundamental assumptions underlie our discussion of software selection:

1. you have a goal in mind that you expect the software and your Apple //c to help you accomplish;
2. your time is limited and valuable; and
3. you have a limited budget for computer software.

These assumptions will not apply to everyone. If you are a computer hobbyist, you may enjoy spending hours learning to use many programs, keeping careful notes on each of them. Or perhaps you can afford to buy hundreds or thousands of dollars worth of programs to compare. But if you are like most Apple //c users, your time and budget are limited. You will need to conduct a systematic and efficient survey that quickly finds a program that will serve your ultimate purpose at a reasonable price. Your time will then be spent mastering this program and using it to accomplish larger goals.

DEFINING WHAT YOU WANT

The first step in selecting a program is to define what ultimate goal

Literally hundreds of programs are available to run on the Apple //c.

you hope to accomplish using your Apple //c. For example, you may want to reduce the time it takes to keep the books for your small business, or to keep the minutes of club meetings. Or, you may wish to improve the accuracy of your budgeting efforts or your child's spelling or school work. Perhaps your goal is to learn about or use spreadsheets or databases.

Once you have defined what you hope to accomplish with a program, you can identify the features you will need. For example, if you plan to improve your child's homework scores by eliminating

misspellings, you may want a word processor and a compatible spelling checker. Or, you may want an entertaining spelling bee game.

For each type of software a number of features are commonly supported. The following chapter defines the features that you will want to consider when selecting software. In addition, you will want to consider whether the program you are seeking will need to work in conjunction with one or more programs. For example, you may need to incorporate data from a spreadsheet or graphics package into reports prepared on a word processor. In such cases, compatibility or the ability to share data between programs may be a major factor in selecting a program.

An excellent way to start your search for a suitable program is to separate the features for that type of program into three groups, based on your particular objectives in using the program. These three groups are:

1. features required to achieve your goals;
2. features that might be useful but are not required; and
3. features that would be a nuisance or dangerous (for example, the ability to erase all fields with a single keystroke might be a feature you do not want in some applications).

Once you have sorted the features in this way, you can quickly eliminate those programs that lack essential features or contain objectionable features. We hope at least a few programs will survive this first cut to be further compared, based on features that are useful but not required, or on the more general criteria of quality and price. If not, see "Increasing the Number of Candidates" below.

Another important aspect of selecting software is determining where to gather information required to make a selection. One important source of such information is computer or software stores. Another source is computer magazines or software guides. Another excellent source is a local Apple users' group. Be sure to consult all of these sources. If necessary, call the vendor of the products you are most interested in.

GENERAL CRITERIA FOR QUALITY PROGRAMS

There are some general criteria for quality that apply to all types of programs. There are also criteria applying to only one type of software, specific criteria. Because features and specific criteria vary so

widely from one type of software to another these are defined and discussed in Chapter 10 for each specific type of software.

We have identified five major general criteria for evaluating and comparing software of all types:

1. price/performance
2. ease of use
3. vendor support
4. documentation
5. error handling

These general criteria are highly interdependent and often closely related to specific criteria for a type of software and even to features. Still, a coherent discussion of the selection process requires that these interrelated criteria be artificially separated and defined. These general criteria may be defined briefly as follows:

1. Price/Performance The main issue in performance is speed. Does the program perform functions rapidly, or is a five-minute wait typical? Allowance must be made for the volume and complexity of the work you have requested. A database search will not be instant with any program, but some programs may perform the same search ten or a hundred times faster than others.

There are other factors to consider in performance. How effectively does the program use diskette space (that is, how much data can it store in a certain amount of diskette space)? Does the program require frequent and/or time-consuming maintenance operations such as reorganizing files or building indexes? Does the program go back and forth to the disk frequently or does it primarily reside and operate in memory?

Finally, the issue of price must also be weighed. While many Apple //c programs are reasonably priced, some are very expensive. In the final analysis, price must be weighed against performance, features, and other criteria. A small improvement in performance or features may not justify a large price increase. Similarly, a small sacrifice in performance or features may save you a lot of money. And, as so often happens, price is not always proportional to performance and quality in software. Your rating for price/performance is essentially your subjective evaluation of how a program compares to its competitors when price as well as raw performance are considered.

2. Ease of Use How easy is the program to use in routine and exceptional cases? Are the functions and commands logically organized and named? Are they easy to learn and remember? Are clear prompts and messages provided? Is additional "help" available on

request? Is the overall functioning of the programs clear and easy to follow?

Generally, the more versatile the commands, the more complex they become. Has the right balance of versatility and simplicity been achieved? Some programs simplify commands by allowing the user to use the single letter of a command word. Consider, for example, a program to help you create music. The command PLAY might be programmed with a single letter P. Some programs provide additional features to make editing more convenient. If it is menu-driven, can you move freely between menus or are you required to reboot the program to get back to the main menu?

3. Vendor Support Problems sometimes come up with programs. It is a fact of life, or Murphy's law if you prefer. The key question here is what happens when you call the producer of the software for assistance? Do they have a toll-free "hot-line" for user questions? Do they have support personnel at all? Do their support people understand the program and its potential uses clearly? Can they give prompt and accurate information and advice? Few experiences are more infuriating than calling up the vendor of a software product that does not work as documented and getting a busy signal. Even worse is getting somebody that you have to educate on the product. Vendor support should be part of what you are paying for when you shell out your hard-earned dollars for software.

A related issue concerns payment for updates and backup copies. When a software vendor corrects bugs in its software or makes minor improvements, these should be available to you free as an exchange for your existing disk, and a very small service charge. If a new disk is mailed, a charge for the media is reasonable. A company that demands that you pay a substantial part of the original purchase price to get its mistakes fixed, however, should be avoided.

Some software is copy-protected by a scheme that prevents you from making a backup copy. This is a reasonable step taken by the company to protect their investment in the software. But the company must provide you with at least one backup copy at minimal cost. Otherwise, if anything should happen to your disk in the middle of a time-critical project, you would be out of luck! A few companies even charge full price to replace a damaged original copy of the program. If waiting a week or more for replacement software does not fit your schedule, be sure you can make your own backup copy or get an inexpensive backup from the vendor.

If you find a vendor to be excellent in these respects and like what you have seen of their output, consider their other products and new releases. A vendor that has brought you good products and service in

the past is likely to continue to offer good products and service in the future.

4. Documentation Is a thorough and readable manual or set of instructions included with the program? Is it well organized with a good index and quick reference guide? Are on-line prompts and help features available? Are these available at different levels of detail and simplicity that can be selected by the user?

5. Error Handling If anything goes wrong (out of memory, out of disk space, file not found, diskette not in specified drive, wrong key pressed, and other problems) does the program provide a helpful message and allow for a simple recovery without loss of data? Can the effects of a major and erroneous action (such as deleting the wrong records) be easily reversed? Can errors in data entry (wrong data, wrong menu item selected) be easily corrected? Can fields with errors be edited rather than retyped? Does the program automatically provide a backup of your data? Does it require verification before proceeding with a potentially disastrous action (such as deleting a file)? These issues are extremely important in determining the overall value of a software package.

THE SELECTION PROCESS

You probably do not have time to carefully examine each of a large number of programs of a given type. Nor would you want to clutter your mind with all the special commands and options of all these programs. The first step in the selection process might well be to eliminate from consideration as many programs as possible. Programs that lack features you have identified as necessary for your intended use may be eliminated. Similarly, a program that offers objectionable features may be eliminated.

We hope at least a few programs will survive this first round of eliminations. If not, see "Increasing the Number of Candidates" below. From among the survivors, pick the ones that get the best reviews and select a small number that you can realistically expect to examine in detail. You may want to write to the manufacturers of these programs for more detailed data. With some perseverance, you may be able to try out all of these at a retail store and make your final decision based on first-hand use of the programs. You may also want to compare notes with other users of the final contenders by talking with friends and user group members who have used the programs.

INCREASING THE NUMBER OF CANDIDATES

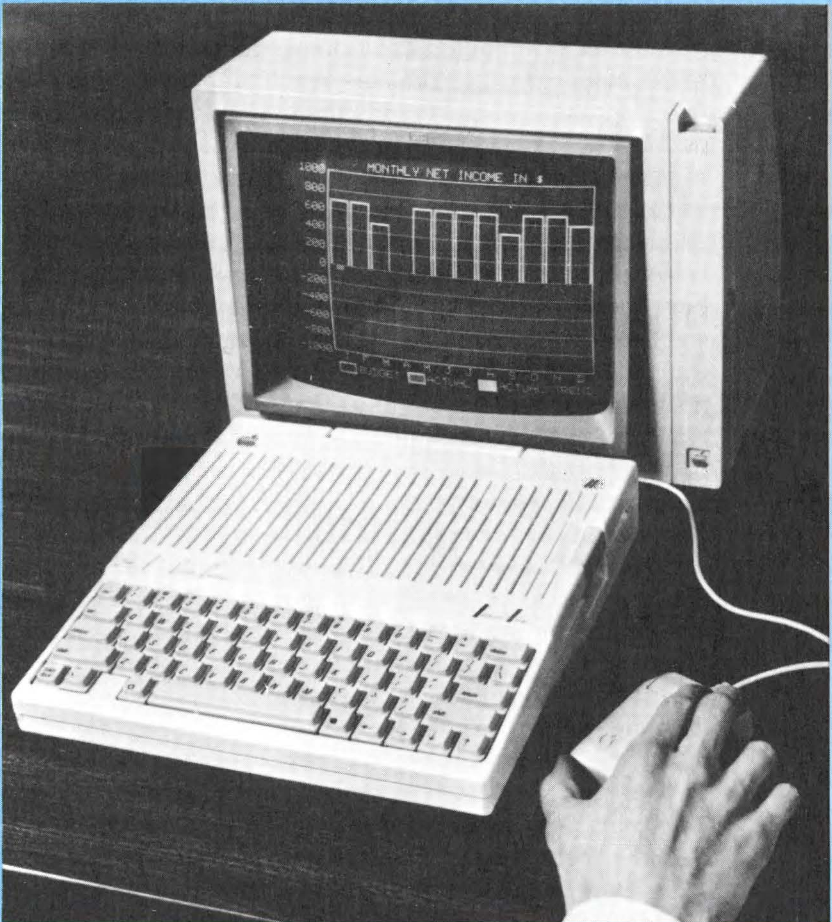
Eliminating all the programs of a certain type that clearly fail to meet your needs may leave few or no candidates. In this case you have a number of options:

1. Re-examine your list of required or objectionable features to see if you can be a little less demanding.
2. Consider stretching your budget to include some of the programs eliminated as too expensive.
3. Consider adding some features to existing programs. This may be easy or may be quite involved.

We hope you find Apple //c programs that you will value as much as we do ours, and that this book will serve you well in your quest for useful programs. Happy hunting!

10

Major Types of Applications Software



EDUCATIONAL SOFTWARE

When hearing about computer applications in education, many Apple //c owners who have children will immediately think of purchasing educational games for them.

Some of you who are educators may even remember the English and arithmetic tutorial programs that were the backbone of computer-assisted instruction a few years ago.

Those of you who are neither parents nor teachers may not be convinced of the value of educational programs. It is not surprising. Most of us have limited exposure to the uses of computers, especially in education. Just a few years ago computers were very expensive. Only big corporations could afford them, and few schools expressed any interest in teaching students with computers. Because of the high cost of setting up the hardware and the paucity of quality software, educators did not suspect the overwhelming effectiveness of computers as tools for learning.

Recent advances in large scale integrated (LSI) circuit technology have brought computers closer to home by making possible very personal, affordable, and portable computers. The Apple //c is perhaps the most outstanding example of quality and power combined into a portable and even friendly package. The Apple //c's application in our daily lives is only limited by our imagination, not the cost of the hardware or software. It is now possible for us to set up a powerful and versatile Apple //c system and use it for a variety of applications at home, in the office, and at school.

Education may be one of the most important applications of the Apple //c. It encompasses all age groups and all levels of interest. The computer provides a self-paced and interactive learning environment. Learning with computers can also be fun. In fact, most of us find it hard to associate the idea of "study" with computer-assisted learning.

The role of the Apple //c in schools is not limited to educational tutorials and games. It is capable of performing a wide variety of tasks outside the education arena and many educators look forward to using these outstanding transportable computers for applications related to the actual running of the schools.

These applications may be considered under the following headings:

Word processing Already one of the most popular applications of microcomputers, word processing is useful for any organization that does a lot of writing or has a heavy correspondence load. In schools, word processors may be used for interdepartmental corre-

spondence, preparing lessons and quizzes for students, and generating research manuscripts.

Information management Schools have to constantly deal with maintaining student records, grades, personnel records, library catalogs, and many other types of information. A computer-based information management system (database manager) is useful in processing these and many other records with ease, speed, and accuracy. With its built-in diskette drive, the Apple //c brings serious database management into the world of truly portable computers for the first time.

Accounting This involves managing finances. With constant budget cutbacks, school administrators need to maintain accurate financial records so that they may implement tuition increases and fund-collection drives (if need be) to keep the school running. Accounting through a microcomputer not only helps them keep an accurate record of their finances, it also participates in generating payroll for employees, financial aids for needy students, and funds for affiliated organizations (such as the student associations and others). Many accounting programs provide forecasting capability and graphic displays of numerical data.

COMPUTER-ASSISTED INSTRUCTION—TUTORIALS AND GAMES

The concept of computer-assisted instruction (CAI) by using tutorials and games is not new to educators. It has, however, acquired a new reputation for practicality in recent years due to the falling prices of personal computers and maturation of the software market.

What Is So Special About Computer-Assisted Instruction?

First of all, it is individualized instruction. The subject material to be learned is preprogrammed into the computer and is presented to the student in a systematic and logical sequence. The computer systematically branches to various lessons, depending upon the student's performance.

CAI is interactive. The computer presents tutorials in the form of lessons as the student reads and learns. It presents exercises and the

student responds with the answers. The program progresses to the next lesson if the answer is correct or branches to a remedial lesson if the answer is wrong.

CAI is self-paced. Students can sit for as long as they want and progress at a pace at which they feel most comfortable. Many find this approach to learning very convenient and productive.

And last but not least, CAI is fun and interesting—and it is another way you can expand on the potential applications of your favorite micro—the Apple //c. Many educators feel CAI can play an extraordinary role in your child's development by teaching the child to enjoy learning. Encouragement to learn is an important feature of any well-designed educational program. Learning can be fun only if the fun is built into the computer program. This is far more important than the particular information target in a CAI lesson.

There are many aspects of computer-assisted learning. Perhaps the most important at home or at school are tutorials and drills. The computer continuously stimulates the student— visually with graphic images, auditorially with music and sound effects, intellectually with problems and exercises, and even the body is involved through typing answers. The material learned in the tutorials may be reinforced in the student's mind with frequent practice problems, quizzes, and drills.

An advanced feature (unique to computers) of CAI is simulations. Computers are able to imitate a real-life situation or condition. Whether you are an airline flight trainee or a biology student learning population migration, with computer simulation you can test the value of various hypotheses, theories, and so forth. You can use those factors and study the effects of the changes. Simulation lets you develop models that tend to change the same way as a real situation. Simulation has wide use in schools as well as professions such as engineering and business.

An extension of tutorials and simulations is educational games. The distinction between a tutorial and an educational game is quite arbitrary. It is easy to think of a tutorial or a simulation as a game if it meets any of the following criteria:

- offers a reward or incentive every time the student picks the correct answer;
- maintains a score of the student's progress;
- offers entertainment by using graphic images and sounds;
- requires input from more than one student;
- induces a sense of competition among the player(s).

On the other hand, a game can be a tutorial if it serves to teach something while the student plays it. We usually think of a game as

having some form of entertainment quality. Tutorials, on the other hand, may be interesting but not entertaining.

Criteria for Evaluating CAI Software

An extensive collection of tutorial programs and educational games is available for the Apple //c. These vary in subject matter, audience, quality, and approach. For example, one program may be a series of tutorials in French for high school students and another may be a game designed to teach preschool math to young children.

In order to make intelligent decisions when faced with such a diverse collection of educational software, you will need to keep in mind some criteria common to all those programs. These criteria are discussed in detail below.

Tutorial Design and Implementation

The effectiveness of a program in providing an environment for self-instruction depends primarily on the way it presents the material to the student. The course material may be divided into many small, logically linked sequences of lessons. Each of these may be presented to the student in a sequence determined by his or her level of performance. The program may also present a set of exercises at the end of each section. If the student provides the correct answers, then the program proceeds to the next lesson in the sequence. If the student provides wrong answers, then the program branches off to a remedial lesson or the help section where wrong answers may be explained. Drill and practice exercises are also often built into the programs to reinforce material learned from the tutorials.

A similar format is used in educational games, except that the emphasis is on scoring and receiving immediate reinforcement by answering correctly. These games prove to be as effective as the tutorials; however, they may not be designed as a linked sequence of lessons.

Interest Level

Even if a program is very sophisticated in design and implementation, if it does not provide an interest-filled environment for the student, its effectiveness is considerably reduced.

Interaction

What makes CAI tutorials more effective than traditional classes as teaching tools is the automatic interaction and the feedback that they can provide to the student. The computer acts as an audio-visual device in which the program displays the information in the form of graphic images and text on the screen. It presents the information and frequently tests the student on that information.

It also responds to any queries presented by the student. It is responsive to the student at his or her level of performance. This one-to-one (student-computer) interaction seems to be very effective in helping young children develop reading and arithmetic skills.

Evaluation Summary A program that ends with some kind of evaluation of the user's performance will be beneficial to the student. Generally, a program will count the right answers given and normally only the first answer given in response to a question will be counted. Certain programs will give the student half credit on the second try or keep asking the same question. After a set number of wrong answers (usually three), the program displays the answer to the student. Other programs maintain and display a running score of the student's right and wrong answers.

EDUCATIONAL LANGUAGES

There are a few languages for the Apple //c (such as Pilot), that have been designed specifically for teachers and programmers interested in developing educational software. These languages are sometimes referred to as authoring languages. They allow teachers to develop effective educational coursework for students without having to learn a lot about computers or programming.

Pilot is designed for sophisticated text manipulation. It is useful for developing tutorials, dialogues, and drills. Pilot takes a handful of simple statements to construct highly interactive dialogue programs. With Pilot, teachers can create their own computer-assisted instruction (CAI) specifically related to their instructional strategies and courseware. Pilot programs provide flexibility in dialogue response so that the students can freely input their response. The program will match that response (even if it is incorrectly spelled) to its own response in the instruction set, thereby allowing "free response dialogue" between the computer and the student. BASIC, in comparison, requires awkward input processing routines for comparing words and phrases that the student might enter. Pilot allows the student to respond in a conversational manner.

In addition to these features, many versions of Pilot also incorporate sprites and bit-mapped graphic displays as well as sound and music synthesis. These features are well suited for educational games and simulations where the program requires an animation effect.

Similarly, Logo is useful for symbol manipulation. Primarily developed for children, it has found a wide range of teaching applications in all age groups. It emphasizes learning through problem solving

(much like Pilot). It is full of simple commands and provides the programmer with a very structured, oriented approach, which also makes it useful for teaching computer programming skills to students. It incorporates a feature referred to as Turtle Graphics. The concept of Turtle Graphics is simple. It allows children and beginning programmers to instruct an imaginary turtle to draw all kinds of graphics on the screen. The turtle lets the child experiment with his or her ideas and then get immediate feedback from the computer on those ideas, thereby learning in the process.

Logo is simple to learn. Besides its use as a learning tool for children, it also has many other features including its ability to explore complex mathematical relationships and ideas. These features allow for its uses in many subjects including mathematics, music, language arts, and the sciences.

Criteria for Evaluating Educational Languages

Although the two languages discussed above (Pilot and Logo) appear to have some differences in design and implementation, many recent versions of these and other languages tend to incorporate any and all features that were mentioned in the context of either of the two languages. These features specifically refer to graphics and sound generation capability as well as text string processing capability.

An evaluation of educational languages for the Apple //c should take into account the following general criteria that are common to these languages (in both design and implementation):

Program Development Many factors are involved in program development. As we have emphasized in the discussion earlier, the primary consideration is the ease with which teachers are able to understand the underlying command structure and syntax of the language, so that they may design tutorials and other computer-assisted instructional software to fit their needs. It is also important to consider how the commands are implemented in a program and how easy it is to develop a program using a string of simple commands. Other factors that are considered here are whether a language has a built-in interpreter or if a program written in that language requires compilation before it may be executed.

Program Debugging This is an important feature of a language, because it provides the programmer with the facility to debug programs through features such as program listing, a trace com-

mand (that executes the program one step at a time), a dump command (that lists the values of the variables and the student's responses in the buffer), and other features.

Informative Error Messages The success of developing a program that works depends a great deal on the help that you get from the language editor and the interpreter. This help is in the form of informative error messages. These error messages will tell you the location (line) in the program where an error occurred and the type of error that occurred. How specifically a program's interpreter relays these error messages to the programmer is the subject of evaluation in this category.

Turtle Graphics Most CAI languages now incorporate graphics commands through the use of an imaginary turtle, which is instructed by the student to move around the screen and draw various shapes and images. Turtle Graphics has become an important part of learning for children, and its incorporation in many of the educational languages enhances their effectiveness and applicability.

Versatility This refers to the flexibility and provisions that a program may provide to extend the realm of its applications. For example, Apple's Pilot for the Apple //c has provisions for integrating a programmer's own machine language subroutines. It also has a record-keeping capability to aid the teacher in maintaining records of students' performances. Versatility also reflects other special features such as controlling joysticks and paddles, accessing the disk drive, and other peripherals.

HOME APPLICATION AND PERSONAL MANAGEMENT

Your Apple //c can be valuable to your family in these times of inflation and high prices. Home management and accounting software can help you establish budgets, monitor expenses, and verify bank statements. Just think how satisfying it could be to send your own computer printout to the electric company notifying them of an error in their calculations.

Generally, home management and personal accounting programs are not meant to find errors in billing. They can, however, greatly simplify the process of balancing checking accounts and monitoring expenses within the home. Most programs require an initial investment of time to set up the program with names, addresses, and ini-

tial balances. Then the user must periodically update balances, enter values from checkbooks, cash receipts, monthly charge card payments, and other information. The real advantage of home accounting programs is to let the user see his or her immediate financial situation. It also allows the user to see exactly where income is being spent. Finally, the user is able to draw realistic expectations about financial goals. When records of expenses and savings are kept for a period of time, it is possible to project future savings and expenses. When considering the purchase of a new car, for example, a computer printout of current and projected auto expenses may help you make a financially sound decision.

Also, there are programs that have nothing to do with household bills. Among these are programs for keeping track of birthdays, anniversaries, and other dates, recipe programs, and home inventory programs.

Criteria for Evaluating Home Applications and Personal Management

Configurability There are a number of checkbook programs that monitor your checking account. They may generate a formatted printout of some type. Make sure that the program will work with your printer.

You need to consider how flexible a program is when first setting up an accounting system. If you have more than one savings or checking account, will the program accommodate separate entries? Maybe you have a dozen credit cards and are making partial payments on each. Can a program include all of your cards? Any limitations of a program concerning size and flexibility will be listed under configurability.

Usefulness With each application we must consider how useful it really is. Does it save us time or does it take more effort to enter the data than is justified by the result?

Generally, menu-driven programs are desirable. They simplify entries and make the program more time effective. You want a program that is versatile, one that can be changed as your financial affairs change. Perhaps you need to close out a checking account or you decide to send in one of your charge cards. Can your system be changed or do you have to start all over again. You want a program that allows you to make mistakes. Before deciding to use a program, make sure you can go in and edit incorrect values without having to re-enter all the subsequent entries. It is nice to be able to insert little

memos along with value entries. Memos like "plumbing repairs" are not necessary for keeping track of expenses, but may be handy at the end of the year when figuring out tax deductions.

Some home finance programs go beyond the checkbook into the worlds of credit, interest, and depreciation. Even a home computer is hard pressed to keep up with fluctuating interest rates, and straight line versus accelerated depreciation. You will want to know how your computer handles those problems. How are transactions handled? If you make a payment on an established bill, will the amount in the checking account automatically be deducted? If you pay exactly $600 each month for rent must you re-enter the amount each month or is it handled automatically? Can you split a transaction? Sometimes a check is made out for a single amount but is intended to cover three or four items. Can your software split the transaction?

GAMES

With the Apple //c you can bring the excitement of arcade games into the home—and take it with you wherever you go! There are three different types of computer games: adventure games, arcade games, and strategy games.

Adventure Games

Adventure games create settings with messages that appear on the screen. You may find yourself in a medieval fantasy where dragons and mythical beasts may appear at any time, or on a planet in another galaxy. The object of the game may be to find a treasure or defeat a monster. Or you may acquire points by overcoming obstacles as the game progresses. In a typical game the computer might initially display some text on the screen to describe your location or predicament: "A swollen river—and beyond, a winding road." You enter into the keyboard your response: "I swim the river." The computer will come back with the result of your action: "A shark just took a bite from your leg, Gimpy. Try again." Usually the computer is very rude. You must rely on logic, deductive reasoning, and previous experience when playing these games. Anyone knows you should have made a raft with the lumber you encountered three statements earlier. Usually you will encounter clues along the way.

In some of the games the player actually assumes an identity. You may be given a list of virtues and abilities such as strength and wisdom, or you may receive some magical powers. (Think how much

easier it would have been just to fly over that river.) Documentation is very important in these games. Not only does it give you your objective and some idea of the vocabulary you must enter, but often it gives you hints about how to overcome the various demons and difficulties. Occasionally, there is a map to reveal the locations of various treasures. It may be helpful for you to construct your own map as you progress through a game. There are generally some restrictions on the words you can enter into the computer, but you may be surprised how much the computer will recognize. If it does not understand a word it will ask you to restate it. The computer will even keep track of your score.

Some adventure games enhance the challenge with graphics and sound. You may see a high-resolution image in the form of a dragon moving across your screen or hear the creak of a door as a pirate liberates you of your gold and silver. Some of the newer games have incorporated arcade-type action by using a joystick or game paddle to control your alter-ego on the screen and to interact with your adversaries.

Adventure games may take hours or longer to complete. A very useful feature is the ability to store the events of the game on diskette. You are then free to turn your computer off and return at another time to pick up where you left off.

Adventure games are always lengthy and seem to appeal more to people interested in the challenge of puzzles rather than the challenge of Pac-Man. They attract a distinctly different type of afficionado but have a following as faithful as those dedicated to the arcade games.

Arcade Games

These are the action games where you may find yourself shooting aliens or maneuvering a lunar landing craft. You interact with the screen using a joystick, a paddle, or in some cases the keyboard of your Apple //c. These games can take full advantage of the Apple //c's high-resolution and the new super high-resolution (full 16 colors) screen as well as either an external speaker or the built-in headphone jack and volume control of the Apple //c.

There can be no logical, orderly pattern for evaluating arcade games. We are convinced they appeal to some primal passion that no one has been able to identify, but the qualities of graphics appeal, interest level, sophistication, and challenge can be discussed. We are sure the controversy over their effect on children will last longer than the games themselves. Some will argue that they are a meaningless waste of time, that they distract children and adolescents from more useful pursuits. But as one arcade fanatic put it, glancing up

from an intense game of Zaxxon, "They build terrific eye-hand coordination and prepare me for interplanetary travel."

Strategy Games

These are the chess, blackjack, and poker type of games. They are often computer versions of more traditional games. Playing against computers can have some distinct advantages over playing against people. Often you can program the level of difficulty. You can find the perfect challenge for your level of expertise. Playing against a computer offers you a much better learning opportunity. You can play at your own pace and avoid those snide remarks like, "Are you going to make that move today, or should I come back tomorrow?" You can play strictly instructive games, quitting anytime you want—generally when you are losing. When learning a new game the Apple //c provides you with an understanding teacher during the wrong-move stage.

Specific Criteria for Evaluating Games

Graphics Appeal How effective is the visual presentation of the game? Does the program make effective use of color? What type of animation does it contain? Do the little men simply move across the screen, or does the program use refined animation techniques so the men seem to walk across the screen? Remember, some adventure games are strictly text messages and do not intend to display graphics. A newer type of graphics, making use of 3-D, is nearing the completion stage. You can wear special 3-D glasses and view the game in three-dimensional perspective.

Challenge How difficult is the game? Is it so easy that the game quickly becomes boring or so difficult that you lose interest due to frustration? Can you change the level of difficulty? Some games allow you to set the level of difficulty in the beginning. Some games become increasingly more difficult as you play. Does the level of difficulty change with the score that you achieve?

Interest Level Can the game captivate the player? Does it use the same theme used by a thousand other games? (We are getting pretty tired of alien invasions.) In playing the game are there a variety of alternatives? In adventure games, this might mean the number of rooms that we could enter or the number of different adversaries we might encounter. In arcade games, it might refer to the number

of different screens that appear. Are there any novel approaches that have not been used in countless other games?

Use of Sound

Sound in a game may range from a little beep to a baroque fugue. Is it possible to generate an infinite number of sound effects, including creaking doors, wind, and, of course, laser bursts? How effective are the sound effects? Are they realistic? Do they vary throughout the game? A pretty minuet of two bars is not so pretty after an hour or so.

Sophistication

This term can be used on all games but is particularly applicable to strategy games. Just how close a computer chess game approaches the real thing relates to its level of sophistication. Does the computer make intelligent moves or do they seem to be random? Can you do everything in the computer game that you can on the blackjack tables of Las Vegas? How long does the computer take to make its move? We may want to take a couple of hours to make our own chess piece move but we do not want to wait that long for the computer to make its move.

MUSIC

Our perception of sound comes from a complicated auditory system of bones and nerves within the inner ear. The ringing of a bell or the blowing of a whistle generates vibrations that are transmitted through air to our ear. The eardrum and bones of the inner ear begin vibrating to generate the sound that we actually hear. The Apple //c can also generate vibrations that are transmitted through the air creating music, sound effects, and even primitive synthesized speech. High quality sound can be produced through either an external speaker, for shared listening, or the built-in headphone jack with volume control for private enjoyment.

As the speaker vibrates it compresses the air in vibrating patterns that are transmitted to our ears. The shape of a sound wave also affects our perception of a sound. The volume of a tone changes from the moment we first hear it until it dies away. If we were to isolate a single tone we could distinguish four separate sections to it. Then we could locate that part of the tone as it went from zero volume to its maximum volume. This is referred to as the attack. The volume will then drop off to a midrange value that is called the decay. How long the tone remains in this midrange is called the sustain level and how quickly it falls to no volume is called the release.

There are several types of software available to produce music on the Apple //c. Some allow you to create music without knowing pro-

gramming at all. Some turn the keyboard of the Apple //c into a piano keyboard with different keys producing different notes. Still others create music with straightforward English commands rather than through BASIC programming. Many of these programs enhance the sound with graphics. Some simply play one or a small selection of tunes for your amusement.

Specific Criteria for Evaluating Music Programs

Quality of Sounds

The prime consideration in the development of sound is the quality of the sound produced. When you are using additional commands to produce music, the quality will depend largely on how versatile those commands are. Can you alter a staccato effect or is it an all or nothing proposition? How pure is a tone produced? If prestored tunes are played, how good is the fidelity? There is a certain limitation to the quality of the sound determined by the Apple //c and the speaker system used. Within that framework the quality of the tone is largely determined by the commands themselves.

Ease of Use

Generally the more versatile the commands of a music program are, the more complex they become. Has the right balance of versatility and simplicity been achieved? Some programs simplify commands by allowing you to use a single letter of a command word. The command PLAY might be programmed with the single letter P. Some programs provide additional features to make editing more convenient. Developing the "right" sound is often a trial and error effort. It simplifies the development process to proceed through a program line by line in a "stepping" process rather than having to go through the whole program each time.

GRAPHICS

Graphics programs have a wide variety of applications for the Apple //c, ranging from amusement to business to education. In evaluating graphics programs, you will want to consider such factors as visual complexity, resolution clarity, color tone, and graphic appeal. Visual complexity denotes how elaborate and refined the designs are. Resolution clarity refers to the sharpness of the image on the screen. High resolution implies clear, precise images whereas low resolution suggests less-detailed impressions.

Color tone is a broad term suggesting the visual appeal of multiple colors and contrasting textures in the graphics display.

Graphics appeal denotes the artistic or stylistic merit of the onscreen display. Naturally, such a judgment is somewhat subjective; your opinions may differ from ours.

Specific Criteria for Evaluating Graphics Programs

Graphics Quality When you buy a graphics program, you may be interested in creating graphs and pie charts for your business, sprite animation for your games, engineering designs for school, or just plain doodling for fun. Whatever the interest, you will want the program to enable you to create high quality graphics. Some programs offer a number of features and functions more than others to facilitate this. For example, you can instruct a program to draw a geometric shape (a circle or a square) on the screen and fill it with a specified color using simple commands or function keys. Or you may use a mouse to create free-form drawings and shapes.

Similarly, some graph generator programs will offer higher resolution than others and will let you create many different types of graphs and charts using simple commands and data input statements.

Print Function The display of the graphics drawings on your TV or monitor screen is often not sufficient—especially if you want to create a report using computer-generated graphics. Surprisingly, many graphics programs do not offer any facility to allow you to print the graphics designs on paper. Programs that do offer this feature restrict the facility to one or more particular brands of printer. For that reason it is very important to know exactly which printers a particular graphics program is capable of using. Some graphics packages for the Apple //c are designed to work on the various Apple printers or the Apple plotter. If it is possible to configure the package to work on any other printer, you will have more flexibility in using the package on different hardware configurations.

PROGRAMMING LANGUAGES

The Apple //c has Applesoft BASIC built in. When you use BASIC you can enter any of the BASIC commands and the computer will execute the specific command. You can enter a number

of these commands, preceded by line numbers, then type RUN followed by a RETURN, and the commands will be executed one after another until the program is completed. BASIC is a fairly universal language among microcomputers. Different computers may alter a command or two, and some provide additional commands that would not be recognized by your computer, but essentially the rules and approach to BASIC programming are the same from computer to computer.

There are a number of languages other than BASIC available for use on the Apple //c. Each language has its own set of commands and rules for implementing these commands; and each language has its own advantages and disadvantages. Some are better suited for developing business applications, some for engineering, and still others for graphics development. Each language will include a group of programs that are similar in nature. There will be some type of editor program that will allow you to enter commands into memory and to edit those commands. Languages such as BASIC, Pascal, and Logo are called high-level languages. They cannot be understood directly by the computer, so they must be converted into a form of language the computer understands. This is usually done in one of two ways. BASIC uses the interpretive method where each instruction is interpreted (changed to instructions the computer understands) as the program is executed. The second method is where the high-level language is changed into the language the computer understands before the program is executed. This is called a compiled or assembled program. A complete development system may also include some type of debugger program.

Learning a programming language requires a considerable amount of time and study. You will certainly want to do some additional reading before you undertake such a project.

Programming Aids

While the Applesoft BASIC built into the Apple //c provides a number of helpful editing features, there are many additional features that Applesoft BASIC programmers yearn for. The response to this need by the Apple software industry has been a number of very helpful tools to aid the BASIC programmer. A number of programs are available to compile BASIC programs for speed, to provide superior listings, to cross reference all variables, to "scrunch" down the program to save memory, and so forth. There are even some programs to allow structured programming in BASIC.

UTILITY PROGRAMS

Many Apple users who were frustrated by the lack of a feature they needed found a novel solution: write a program to provide that feature. Many of these "utility" programs became commercial products for the Apple market, while others were released into the public domain. Some of these programs provide simple extensions of existing DOS functions. KRUNCH, for example, gives a sorted directory listing of a diskette. KRUNCH not only sorts the files alphabetically, it also removes the deleted files permanently from the DOS 3.3 directory. The main benefit from using KRUNCH is that if you have a catalog program that displays deleted files and you are like the author and hate cluttered-up catalogs, you may delete them all in one fell swoop. These extensions to APPLE-DOS and ProDOS can make day-to-day use of the Apple much easier.

There are many other utility programs available to "soup up" the Apple's performance.

BUSINESS AND ACCOUNTING PROGRAMS

Today's businesses require an accurate record of all financial transactions. This record is useful in verifying the financial condition of a business and may be required for bank loans, tax audits, and numerous other routine activities. Such records can also be useful in identifying weaknesses in a business before they become fatal. Originally these records were first entered into a journal and then transferred to a ledger. This ledger was broken down into categories such as cash, office expenses, and other categories. Actually, the ledger was subdivided into three main categories: accounts receivable, accounts payable, and general ledger. The result of every transaction would affect the ledger, and the true economic condition of a company could easily be assessed from the ledger. There was a great deal of repetitive action involved in updating the journal, transferring to the correct ledger, and verifying that all balances were correct. Repetitive action is exactly what a computer does best, and some of the first business applications were in the field of accounting.

Accounts Receivable Few transactions in the marketplace today involve cash on delivery. Whether it is a coat from Macys or the replacement of a filling in a dental office, payment for the product or service is usually made after the fact. A receivable account is

made by a business on every customer that qualifies for credit. Each time a customer makes a purchase, an entry is made showing the date of the purchase, the amount due, and the terms of the sale (to be paid in 30 days, and so forth). When a payment is received, another entry is made in the account along with the date and the amount credited to the account. In this way, it is possible to look at a receivable account and determine how much is owed to your business and when the balance is due at any time. By adding the totals from all of the receivable accounts, you can get an overall view of expected revenues. These last couple of items are quite important from a business point of view.

This kind of information can help you decide whether or not it is wise to be spending money at this time. It can also help you decide whether it is a good idea to continue to extend credit to a customer or perhaps hold off until the customer has made good on what you are owed. Banks view accounts receivable as a type of asset, and such records may be necessary when applying for loans. A good accounts receivable program makes accessing this kind of information quick and convenient.

An accounts receivable program is initially set up to include all existing customers to whom you wish to extend credit. Each time a new customer qualifies for credit, that name is added to the list. You will then be prompted to furnish information on the new customer, including the name of the firm, the address, and phone number. Depending on the particular program, you may be asked to furnish the credit limitations and the acceptable terms for payment. Periodically, someone in your firm will post all payments received into the accounts. That is, they will record each payment and make the appropriate entry in the appropriate account. The advantage of using a computer to handle receivable accounts becomes apparent as the list grows. Accessing information as to who is late on payments and who is approaching their credit limitations remains a convenient process.

Accounts Payable A business keeps track of the amount it owes each of its vendors in payable accounts. Any purchase on credit, whether it is office equipment for its own use or merchandise for re-sale, is kept track of in payable accounts. Each time a business makes a payment to one of its vendors, it is recorded in the appropriate account. Essentially the same type of information kept in a receivable account is kept in a payable account, including the name of the vendor, the address, the phone number, the initial amount, and the terms of the sale. Again, new vendors can be added to the list at any time. As with receivable accounts, payments are posted into the approprate accounts. Remember, this is a record of money owed by your business.

It is often necessary to access information from payable accounts to resolve any discrepancies between your records and those of the vendor. When the list of payable accounts becomes long and the number of time payments becomes unmanageable, it is extremely helpful to have a program that can sort through the list, picking out those accounts where a payment is due immediately. This kind of information is vital when management attempts to structure budgets and plan expansion.

General Ledger A general ledger keeps track of all assets, liabilities, and capital. The amounts owed in payable accounts is totaled and entered into the general ledger as a single entry. The amounts owed to your business from receivable accounts is totaled and entered as a single entry into the general ledger. In addition to those entries, there are separate entries for cash and checking accounts, office equipment and supplies, merchandise, and repairs. A record of anything of value is kept in the general ledger. The general ledger lets a business get an overview of the economic health of the business, whether it is making money or losing money, and what the major sources of gains and losses are. Total assets and liabilities should balance out in the general ledger. Any discrepancies indicate an error in calculations or misuse of funds.

A general ledger is set up when a business begins operating and is updated periodically when transactions are posted. A general ledger software package allows the user to view and to copy individual accounts in the ledger and to summarize all accounts to receive an overall picture. A general ledger package must be able to interact with an accounts payable and accounts receivable system. If those two kinds of entries are few in number, they may be kept in a book and entered into the general ledger periodically. If your business has accounts receivable and accounts payable software, you will want a general ledger package that can interact with the two.

Inventory An inventory software package keeps track of a business's saleable items. There is generally a delay between the time products are ordered and the time they arrive. This, coupled with the fact that some products sell faster than others, may result in a surplus of some items and a shortage of others. Inventory software can help a firm keep track of available inventory and avoid a surplus or shortage of products. Generally, when an inventory package is implemented, the user is prompted for specifics on each item. There will be questions regarding the description and initial quantity. The package will request or furnish a specific product number for each item. New items can be added to the list at another time. Each time an item is sold or ordered an entry is made. When a product arrives a

similar entry is made. An inventory package allows a business to have immediate access to information about quantities on hand and the volume of sales on particular items. Some packages use various statistical methods to determine the optimal numbers of an item to keep on hand. They will alert a business to reorder a product.

Payroll Another accounting responsibility involves issuing employees the correct salary. Deductions can become an accountant's nightmare. Generally these deductions include state and Federal withholding that varies, depending on the wages and state. There are often local taxes, social security, and in some cases benefit deductions. Of course wages and deductions can change at any time. It is important to have a software package that is flexible enough to allow for these changes. It is useful to have a package that will print the individual employee's check and maintain an active record of all wages paid. At the end of the year you will want a package that can produce information for W-2 forms. Generally payroll software will prompt the user for all the information necessary for setting up a payroll schedule for an individual. The user will be asked for the individual's name, address, and wage, as well as the percentages for Federal, state, local, and miscellaneous withholding. Once the payroll software is initialized, the user can simply enter the hours worked and the dates. The software will do the rest.

Financial Utilities In addition to the accounting programs mentioned, there are a number of utilities that aid in specific financial calculations and planning. There are often repetitive calculations specific to a particular type of office. You may require amortization or interest rates on various loans or depreciation curves for assets. Essentially, financial utilities are miscellaneous financial application programs. Another category involves making projections based on historical data.

Adaptability Although there are accepted accounting practices used in most businesses, each business has its own specific needs. It is desirable to be able to alter an accounting package to meet those specific needs. Will the software accommodate the number of accounts or the amount of inventory in your business? Can you alter the reports generated to meet your needs? Will the format of the invoices and checks that are generated on accounting software comply with the invoices and checks you already have? Can you change the format to meet future needs? Many payable and receivable accounts must keep track of amounts that are due in 30 or 90 days. This type of accounting practice is referred to as aging, and an accounting package should be able to handle your aging require-

ments. Periodically, a business must sort through all the payments received as well as the payments made and make entries into the payable and receivable sections. This process of posting may be done daily, weekly, or monthly, depending on the volume of your business. Is the posting procedure flexible enough for your business?

Accurate accounting records are vital to a business. A business can become very vulnerable when more than one person has access to the accounting records. Some software packages establish a degree of security by providing password protection for the records. When you set up the account you establish your own password. From that point on you must enter that password to gain access to the accounting records.

WORD PROCESSORS

Word processing is one of the most popular applications of personal computers. A word processing program makes your Apple //c function like a super typewriter. Rather than typing a letter on paper, however, you type it into your computer through the keyboard. The word processor simultaneously displays it on the monitor screen.

The value of word processing on the Apple //c lies in the powerful editing capabilities of a full-featured word processor. Once the text is displayed on the monitor screen, you can move, insert, or delete any portion of the text by using simple commands. In this manner, you may edit several pages of documents when necessary without having to retype a page. With a word processor, you may format your text in any way you would like to see it printed (such as, left justified, right justified, or centered). A word processor automatically numbers the pages of your document. It can also create headers, footers, and line spaces according to your specifications. You can print your documents any number of times simply by typing one- or two-character commands. You can also store your documents on floppy diskettes and retrieve them whenever you wish. This eliminates accumulating loose pages of text during the editing process.

Applications of Word Processors

Before computers became affordable for the general public, word processing was used mostly in business. The "dedicated" office word processing system was specifically designed for professional operators and cost thousands of dollars. Few individuals could justify spending such large sums of money for word processing.

Now a word processing system can be set up on a personal computer (including the portable Apple //c) for a very small investment. This has generated many cost-effective applications of word processing. For example, a small business can use a low-cost word processing system to generate reports for investors or personalized letters for clients. In fact, all business correspondence may be handled by a word processor in a fraction of the time it takes a conventional typewriter. One of the most welcome spots for low-cost word processors has to be among the script writers in Hollywood who spend grueling hours constructing dialogue for television and movie stars. Not only does word processing speed up their production, but it also allows them more time to think and, therefore, increases their creativity. (At least we hope it does!)

It is not surprising, therefore, to imagine word processors as being one of the primary reasons why many people invest in the Apple //c.

Many home computer owners have also started using word processing for applications such as publishing newsletters for amateur, special interest groups (like the Apple users group in your neighborhood), writing term papers for a class, or just sending letters to friends. Some even wind up writing books like the one you are reading, which was edited on a home computer with a word processor.

As Apple //c owners, you are especially fortunate if you want to add word processing capabilities to your computer. Many programs are available for the Apple //c, and in order to choose the right kind to match your business or personal needs, you will first need to develop a good understanding of the concepts of word processing and, second, need to know what features to look for in a particular word processor. This will be discussed in detail under "Specific Criteria for Evaluating Word Processors."

Definition

Word processor was originally a term used for automated, computerized office systems (such as Wang) designed strictly for preparing, editing, and printing business correspondence, corporate reports, and so forth. These systems incorporate a typewriter keyboard, a video display unit (monitor), a letter-quality printer, and built-in software designed for word processing.

With reference to personal computers, the term word processor refers to a program or programs that allow you to create, edit, store, and print textual matter, thereby closely duplicating the functions of a professional word processing system. The obvious advantage is, of course, the low cost involved in setting up such a system for your

personal computer. Also, a personal computer, unlike a dedicated word processor, can be used for many other business and home applications, such as spreadsheets, accounting, information management, mailing lists, educational programming, or games.

Specific Criteria for Evaluating Word Processors

The following is a brief discussion of the criteria that you can use to evaluate the specific programs.

Start-Up Options These are the options presented to you by the program (via the master menu) when you first load and execute the word processor. Through these options, you may configure the system peripherals from within the program. These options may include printer setup, DOS commands, 40- or 80-column screen, and other options.

Printer setup refers to initializing the particular printer that you may be using with the Apple //c.

DOS commands refers to accessing disk functions such as formatting a diskette and obtaining a directory of disk files. Most commercial word processors work in the 80-column mode built into the Apple //c.

Text Entering Text entering simply involves typing text onto the monitor screen through the keyboard. Some programs offer special features that make it easier to enter and format text on the screen. An example would be tab set, which allows you to move quickly to a pre-set position (horizontally or vertically).

Text Editing Text editing is one of the primary functions of a word processor. In the text editing mode, you can display a letter or a report on the screen page by page. By using the cursor keys, you can position the cursor on any part of the text. You can move the text around by scrolling it vertically or horizontally (depending upon the particular word processor). Once you have viewed the text, you can modify it by making the appropriate deletions and insertions. You can erase, rearrange, and copy a line or paragraph.

Some state-of-the-art word processing programs offer advanced functions, such as the ability to "search" throughout the text file for specified strings and "replace" those strings with the corrected ones. You can access all these functions of the text editor in most of the word processors. Each function, however, may require more or less

manipulation on your part, depending upon the sophistication of the particular program.

Text Printing This is the part of the word processor that emulates the functions of a typewriter. It may be considered the most important feature of a word processing program. In the text output mode you can format your text in exactly the way you would like to see it printed. In order for you to accomplish this, most word processing programs offer an array of features. For example, you may be able to set page length (lines), page width (columns), and left and right margins. Some word processors offer advanced features such as automatic page numbering, headers, footers, and forced pagination. You may also be able to send control codes to the printer for accessing alternate character fonts and special printer functions like underlining, boldfacing, superscripts, subscripts, and other options.

File Handling Once you have typed your text into the word processor, you can save it on the Apple //c's disk in the form of files. These files may be recalled whenever you want to view, modify, or print them. In addition to storing and retrieving these files, some word processors include some useful file facilities, such as the ability to link a number of files to form one long document, insert information from one file (fill file or merge) into another, and so forth. These file facilities enhance the usefulness of the program and broaden its applicability.

SPELLING CHECKERS

After spending hours writing a manuscript at the keyboard of your Apple //c, the next step in putting in the finishing touches is editing. This means correcting spelling, grammar, syntax, sentence structure, and other editing tasks. Then you may want to type or print the manuscript to make it visually presentable to the readers. This would normally involve setting the appropriate left and right margins, line spacing, paragraph indentations, and other finishing touches.

If you are using a full-featured word processor on your Apple //c to create your letters and manuscripts, then the task of formatting the manuscript for a presentable report can easily be handled by the word processor. The actual editing process, however, still has to be accomplished manually; that is, you have to comb through the whole manuscript word by word to locate any spelling and grammatical errors. For many of us, this is perhaps the most tedious aspect of producing a manuscript and it involves frequent references to the dictionary and thesaurus.

With the introduction of special programs, referred to as spelling checkers, a major part of the editing process has been considerably simplified. A spelling checker reads your manuscript (a word processor text file) many times faster than most of us are capable of reading. It also simultaneously checks each word in the manuscript against the dictionary that is provided with it. In this manner, you can correct spelling errors at a much faster rate and with considerably more ease than if you proofread the manuscript yourself.

The application of spelling checker programs is, for the most part, limited to just that—proofreading for spelling errors. Very few of these programs (available for the Apple //c) are capable of identifying grammatical errors, errors in syntax, or errors in sentence structure. Moreover, they will not help you make any more sense out of your text. They only correct misspelled words.

The real application of a spelling checker program is for those who are heavily involved in writing or editing manuscripts. If you use a word processor for writing letters to friends or a class term paper, a simple proofreading should suffice.

Functionally, a spelling checker program proofreads by attempting to match each word in your word processor text file (manuscript) with a word in its dictionary. If a match is found, then the word is assumed to be correct and the next word is searched. If no match is found, the word is "flagged" so that you may correct the misspelling if you wish. Since the size of the dictionary is limited, a "no-match" only suggests that the word does not exist in the dictionary. It does not suggest that the word is misspelled. The choice is left to you to either identify and correct the misspelling or pass it as a new word and optionally add it to your dictionary. A limited dictionary of the most commonly-used words in the English language is provided with most spelling checkers. Some also provide a supplementary dictionary to which you may add words that relate to your particular needs. So whether your word processing applications relate to writing scientific papers or business correspondence for real estate clients, you can easily create a supplementary dictionary to suit your specific needs.

Criteria for Evaluating Spelling Checkers

There are several spelling checker programs available for the Apple //c computer. Many of these are marketed as a package with a particular word processing program. Quite often, these spelling checkers will also read text files created by other word processors. When purchasing one for yourself, make sure you buy the appropriate (compatible) spelling checker for interfacing with your word processor.

In evaluating spelling checker programs for the Apple //c, the following criteria should be considered.

Text Proofreading The primary function of a spelling checker is to proofread the text for misspellings. Most spelling checkers will first load the word processor text file from disk into the main memory of the computer. Then, each word from that file is compared to the words in the dictionary. After the comparison is made, the programs will generally display the file with suspect words "flagged" or highlighted for subsequent editing steps.

Text Editing Once a given text file has been read by a spelling checker program, you are given the option of altering misspelled words or inserting new words into the dictionary. The editing process is the key to the efficiency derived from the use of these programs. A good spelling checker will display the "suspect" word or phrase. Many programs display full screens of text with highlighted "suspect" words. This allows you to edit words in the vicinity of the "suspect" word and suggest potential corrections for each word highlighted. It will also support the standard edit keys of the Apple //c for ease of use. After all the words are edited, there should also be an option in the program to save the edited file on disk.

Dictionary The dictionary structure associated with the spelling checker program, the number of words in the dictionary, as well as the additional space available for users to add more words should be considered. Other utility commands and functions relating to the maintenance of the dictionary should also be considered.

DATABASE MANAGEMENT SYSTEMS

Whether you are a businessperson operating a retail store, a hobbyist with hundreds of prized collectables, or a student writing a term paper, you may find yourself in need of a filing system to help organize a wide variety of information (invoices, mailing lists, stamp prices, bibliographies, and so forth). Frequently, you will want to update the information in the files (records) and also print a list of selected items in the file.

Before microcomputers like the Apple //c became widely available, such a record-keeping system commonly consisted of a file of 3 × 5 index cards. Each card contained the desired information about one of a group of similar items (information about a magazine, perhaps, or on a customer with whom you correspond) and was filed in ascending numerical or alphabetical order.

There are several limitations to such a filing system. First, the retrieval of specific records from an index card file can only be made at one level; that is, if the cards containing a mailing list are arranged in alphabetic order by the client's last name, then they can only be accessed by last name. If you wish to access selected cards by a specific city or zip code, you will have to search the whole card file. This also makes record updating very tedious. Another disadvantage is obvious. You cannot, through such a filing system, print out a list of all or selected records.

The card index files are most popular in libraries. Librarians get around parts of the above limitations by maintaining three index files for each publication (that is, subject, author, and title), and by using lots of cheap labor (such as students).

The availability of Database Management Systems (DBMS) on the Apple //c has made it possible to enter and store a wide variety of information on disk files, thus eliminating the need for 3 x 5 card files, large filing cabinets, and hours of labor. Information in these files may be stored in a wide variety of classifications, all of them defined by you. Furthermore, any record or records that you select may be retrieved quickly from a file at any time. Facilities are also available for you to print records in a report format. The fact that these records are maintained on disk files also makes them much faster and easier to update than a card index file. Some sophisticated programs also allow for merge capability (merging records with text files). This feature is especially useful in creating personalized form letters. A calculator function, available in a few programs, even allows you to perform arithmetic and logical operations on records containing numeric data.

Applications of Database Management Systems

The applications of database management systems extend much further than the examples cited above. You may store any type of information that you want by creating your own fields (see the definition below). The program will let you search through that information, sort it in any order, and generate a list of any or all records in that database. For example, let us assume that you have created a database of your customer accounts in the United States. You can now generate a number of lists from the database. You can get a list of your most valued accounts in the U.S., or all your accounts in a specific area.

If you have a database consisting of customers' mailing addresses, then you can get a list of customers in the order of their last names, or a list by a certain zip code. With the merge feature, you can merge each record in the mailing list (containing the customer's name and address) with a standard form letter, that you may create on your word processor, to generate personalized letters.

Another DBMS special feature mentioned above is the calculator function. Using this feature, you may perform calculations on parts of your records. This is helpful when you wish to update the balance of your customer accounts. The calculations may be performed on all or selected records.

In addition to maintaining customer accounts and mailing lists, DBMS are also used for other applications such as stock records, inventories, contract records, student records, sales ledgers, invoices, personnel records, and other material.

Without a computer-based database management system, these tasks may take hours or even days to accomplish manually. Imagine sorting through a list of 1,000 customers to extract those that live in the area code 415, or manually updating invoices or cash balances of 100 accounts. The power of a computer-based database management system is, therefore, in organizing your stored information in any way you want, in a fraction of the time it would otherwise take you. How you use this to benefit your business depends a lot on your understanding of the concepts of database management and a knowledge of how to find the right system to match your specific needs.

Definition of a Database

A database is a collection of information. It is typically a filing system where the information is organized and stored in the form of user-defined structures called records. An example of this is a library card catalog. The catalog may be considered as a database, with each card in the catalog representing a record. The content of each record is subdivided further into data types (author, title, index, and so forth) referred to as fields. A database, therefore, consists of records and a record consists of fields. By definition, all records in a database must have identical data types (fields). Going back to our example of the library card catalog, all cards have identical fields (author, title, and subject), and a student can access a book via any of these three fields.

A DBMS, also referred to as a file management system, is a collection of computer programs that allow the user to create a database and store it in the form of files on a disk.

Criteria for Evaluating a Database Management System

In evaluating a particular database management system for the Apple, the following criteria should be considered.

Start-Up Options When you first load and execute a DBMS program, it offers (via the main menu) a variety of options. Using these options, you may configure the system peripherals from within the program. These options may involve printer set-up, DOS commands, and other options.

Printer set-up, for example, will let you set up the program for the particular printer that you have.

DOS commands refers to accessing the functions and commands of the disk operating system from within the program. This enables you to format a diskette or obtain the disk directory without exiting the program.

File Structure and Specifications The efficiency of a particular DBMS in storing and retrieving data from a file depends primarily upon the file structure used. Random access files provide the fastest data storage and retrieval. Sequential files, however, are only accessible in the order in which they are stored, so the last record entered into the file will be the last record accessed. In order to implement many DBMS functions, such as sorts and searches in sequential files, the data is usually loaded completely into memory. This restricts the size of the file (due to limited memory available in the computer) and makes it less versatile. Restrictions on sorts and searches of large files are not necessarily a function of sequential or random access files, but rather program design—if the program is memory-based or file-based (disk-based). A disk-based system is limited only by the storage capacity of the disk and can handle large files, but since it must constantly access the disk it is slower. Consequently, updating those records is very time-consuming and tedious. Most high quality DBMS use random access files to handle record storage.

Specifications refers to the limitations a program imposes on the file and record structures you can create. For example, a program may allow a maximum of 25 fields per record and 30 characters per field in each record. If you wanted to use that program to record a mailing list of customers, these specifications may be sufficient. But if you want to store the abstracts of magazine articles, you may not have enough space. If you have a specific application in mind, you will find it easier to choose a particular database

management system. Otherwise, generally, the more versatile a particular program is, the more favorable it will be with respect to general applicability.

Advanced Data Handling This refers to features such as "sorts" and "searches" that are available in most programs. The sort feature allows you to arrange the records in your database in a number of ways. You can sort a file in either alphabetic or numeric order. You may use one or more fields to sort your data. For example, if your database consists of mailing addresses, by using the advanced sort feature you may arrange that list in alphabetical order by a customer's last name or city. You may also want to rearrange the same list in numeric order by zip code. The search feature lets you look into your database for specific records. You can define the criteria by using "conditional" statements (such as IF Last Name = Smith OR City = New York) and the program will automatically search for records that match the criteria defined in the conditional statements. Sorts and searches may be performed at one or multiple levels.

Another feature available in most advanced DBMS is the ability to set up calculated fields in your database. This allows you to perform mathematical operations on specified fields of your database file (such as adding the tax to the price of a stock item or averaging student grades). In many cases, you can use BASIC's mathematical operators for your formulas for the calculated fields.

Report Generator A useful function of a DBMS is its ability to generate user-defined reports. These reports may contain a few or all of the records in the file. The reports may be organized as a table or a listing, and the fields may be positioned anywhere on the paper. This flexibility in defining the report format makes a program versatile in its applicability. You can print mailing labels or get a simple listing of a few names and addresses. You can also print selected fields from each record to form a comparison chart. A good database system will support many different types of printer configurations and print formats.

SPREADSHEETS

Imagine sitting down to calculate a household budget for the coming month. You know exactly how much is needed for rent, the car payment, and insurance, but food, auto expenses, and entertainment are all weekly expenditures that you wish to limit to fixed amounts.

The only restriction is that the sum of all expenses must fall within your monthly salary. So you list the different items on a piece of paper and make a guess as to what you will spend for each item. You add up the weekly expenses for each item and then add the monthly totals to find you are planning to spend $75 more than you make.

It then occurs to you that you have been meaning to start that diet for some time now, and the current movies have received the worst reviews you have seen in years. So you change the entries for food and entertainment, reducing the amounts on each just a little, and recalculate. Now you are only $10 over your salary. You toy with the idea of taking the bus to work, whereby you could save $10 a week in parking alone and think of the exercise. Not a chance! You have to draw a line somewhere. And you are totally committed to your savings plan. You trim just a bit more off entertainment. You recalculate and finally you are living within your means.

The process of setting up a budget is very similar to many of the applications in business. Businesses spend a great deal of time adding lists of numbers, making changes in initial values, and recalculating totals. They can project profits or losses on a "what if" basis. "What if" we buy that new computer for $50,000 and avoid making monthly payments to that accounting firm? For a short home budget, it is not all that difficult to erase and recalculate a few times. But when you are working with several pages of numbers, the idea of changing an initial value and recalculating the whole thing can unnerve the soundest of junior executives. It was probably out of compassion for junior executives that the spreadsheet was developed.

Spreadsheets initially display a screen made up of a grid of columns and rows. The entire grid is sometimes referred to as a spreadsheet since this is where you enter and view all your entries. Columns and rows intersect in small rectangular boxes called cells. Each cell can be identified by a row number and a column number. A cursor appears in one of the cells, and you can move the cursor from cell to cell using the cursor controls. You can place the cursor in any cell and enter into that cell words like rent, food, and entertainment, or you can enter values like $57.25 or $681.

You can also enter into cells formulas that can reference other cells. You could have a formula that adds cells A1 to M1. The entry would probably appear as Sum(A1-M1). The spreadsheet would automatically add the values in all of the cells from A1 to M1 and place the total in the cell where you have entered the formula. Do not worry about losing the formula. Each spreadsheet allows you to see the formula again, if you wish, but all you care about is the result anyway. What if you decide to change just the entry in A1? Here is where the power of the spreadsheet appears. You simply change the

entry in A1 and instantly the formula recalculates the totals of A1 through M1.

On some spreadsheets it may take one extra step for recalculating all formulas. In either case, the junior executives (and home managers with complex budgets) are smiling again. Usually, with the use of the cursor controls, your screen becomes a kind of "window"that can move up and down or left and right, to view a larger "page." You can only see one area at a time, but you can simply move your window several times to eventually view the entire spreadsheet.

The same dilemma develops when you want to make a copy of your work on a printer. You cannot print beyond the maximum width of your printer, often only 80 columns wide. The only way around that problem is to print sections of your work. The best plan is to simply design your spreadsheet with your printer in mind.

Many of the programs available for the Apple //c that perform specific operations, particularly accounting functions, are essentially spreadsheets. They may be menu-driven with formulas already entered, and they may meet a specific need for your home or office. A good general spreadsheet can be a very versatile tool with many applications. If you are in sales, you may use a spreadsheet to record weekly sales statistics. Then you can let the spreadsheet compute the monthly sales figures, calculate deductions for taxes, and figure discounts for bulk sales. If you are in management, enter the names of your staff and the hours worked, and let the spreadsheet calculate their wages and deductions.

Criteria for Evaluating a Spreadsheet

The following criteria should be considered in choosing a spreadsheet for your Apple //c.

Configurability As with all software, it is a very tricky business to select the right program for your needs. It is even more of a problem with a general spreadsheet since you may have more than one application in mind. Configurability is one of those buzz words that you would prefer to skip over but it keeps popping up in everything you read about computer software. It has something to do with flexibility. Will your software print to more than one printer? Can you use two disk drives? Can you change the color of the screen? What about the screen itself? Can you alter the design to be more functional or visually appealing? Just what are the limitations on this software package? The answers to these questions all fall under the heading of configurability.

A major consideration in a spreadsheet should be its size. Each name, number, and formula that you enter into a spreadsheet requires additional memory within your computer. Your Apple //c has a finite amount of memory available. Your computer memory size limits the number of entries you can make. It also specifies the maximum number of cells available as well as the maximum number of rows and columns available. It may be necessary to keep all three factors in mind at the same time. If there are a maximum of 1,000 cells (with a maximum of 200 rows and 50 columns), it does not mean you can calculate a spreadsheet with entries in 200 rows and 50 columns at the same time. That would require 10,000 cells (200 x 50). You could have 200 rows and five columns of entries, or maybe 100 rows with ten columns, just as long you did not try to have more than 1,000 cells in use on your spreadsheet.

There are other considerations in designing your spreadsheet. How wide is a column? Many spreadsheets appear originally with columns eight characters wide. But what if a number you want to enter into the column is nine digits long? Actually you can still enter in the complete number; you just will not be able to see the whole thing. Your computer will remember the complete value, and you can add and subtract and do anything your spreadsheet is capable of doing—but up on your screen you will see only eight of the nine digits. As long as you remember it's really $10,000,000 and not $100,000, there should be no problem. Fortunately, most programs allow you to set up your spreadsheet with column widths different from the ones originally displayed. Again, there will be a limit to how wide you can make the columns. In some programs you must decide in advance how wide your columns will be. Some let you alter column widths after you have already made entries in the columns. In some programs you can make a global change of column widths, changing all column widths simultaneously, while in others you will have to change the columns one by one.

Regardless of your application, chances are you will need a printed copy of your work. You want to be absolutely certain your spreadsheet will work with your printer. Common sense tells you that if you have been using your computer with your printer until now, that it will continue to work when you purchase your spreadsheet. Unfortunately no one has been able to program common sense into a computer.

Data Entry and Editing Once you have configured your printer and designed your screen you are ready to start entering data onto the page. Some programs go to great lengths to simplify data entry. They have built-in editing functions that will save you time and aggravation. What if you find yourself entering a list of 30 identical

numbers? How nice to have a spreadsheet that will copy a cell into a range of cells. Even nicer, how about a feature that will allow you to copy an entire row or column? Maybe you do not want to copy a column—you just want to move it to a different location. Is there a single command that can accomplish this feat? On some spreadsheets yes, on others no. A slightly more advanced feature to grasp is a relative move of a column or row. You may have a column or row with formulas that reference other cells. When you move that column or row, you may want the formulas referencing it to be changed to account for the new location. Some programs allow for relative changes of formulas following a move.

When you enter a name or a number, it will automatically be aligned to the right or left of the column. It is sometimes helpful to be able to change the alignment—to align or justify it to the opposite side.

Normally one moves from cell to cell using cursor controls. But when a spreadsheet becomes extremely large, it is handy to be able to jump directly to a specified cell by row and column number.

If you have a list of names along the first row and you start listing numbers below the names, at some point the names move off the screen. Remember, you can only view 25 rows at a time, and to view row 26 you will lose sight of row one. A very useful feature for those of us with a short memory is the ability to fix a row or column so it will not move. With this feature the other rows will scroll under the fixed row and you can make entries in row 190 and view row one. Some programs carry this one step further by "splitting the screen." The program fixes one section of the screen so that the other can scroll under it. And if you wish, you can reverse the operation fixing the second section of the screen and scrolling through the first. At least one spreadsheet on the market allows you to create "windows" on your spreadsheet. You can view a small rectangular section of the spreadsheet and view it beside any other section.

If you are certain of nothing else when working with computers, be certain that you are going to make mistakes and will need to make corrections. How helpful the program is in this process is going to be important to you. Some programs provide you with a little help screen that gives you a brief outline of the various commands and saves you the time of thumbing through the manual. Some programs require that you re-enter an entire name, or number, or formula. Others let you go in and make minor changes to existing values. Can you delete an entire row or column?

When you are finally through entering and editing the data on a page, you are going to want to save it to a floppy diskette. Even the most fundamental spreadsheets will allow you to name and save each of your pages. Some programs even allow you to view portions of two different pages simultaneously. You can hold up last year's

budget against this year's budget on the same screen. Anyone who has had to copy a list of figures from one spreadsheet to another will appreciate being able to place them side by side on the screen.

Calculations The power of the spreadsheet is in its ability to perform mathematical operations on numbers. The Volkswagen of spreadsheets will allow you to add, subtract, multiply, and divide. A Volkswagen is fine until you see a Mercedes. The Mercedes of spreadsheets will find a minimum value from a column or row, find a maximum value, and find a mean value. It will automatically calculate percentages, absolute values, and exponents. The Mercedes is going to let you enter logical operators into the formulas. You can AND and OR different cells to your heart's content or NOT. You can make use of inequalities. If cell A1 is greater than cell A3 then enter 100—otherwise enter zero. Perhaps you are listing sales figures and you offer a discount of 10% when a customer purchases more than $500 worth of merchandise. Inequalities help you set up for this situation. Some spreadsheets will alphabetize lists of names and sort columns of numbers.

All of the mathematical and logical operations take some time, depending on the quantities involved. Some programs allow you to make all the changes to your entries before recalculating formulas. In others, the recalculations are made after each new entry. Some programs are much faster than others. Those written in BASIC are generally very slow, and a long column of formulas may take a couple of minutes to recalculate values. The same column processed on a spreadsheet written in machine language may take only ten seconds.

Report Generation When you are setting out to convince someone of a trend, there is nothing like a computer printout to make it obvious. The printout from a spreadsheet can be very convincing—long lists of names and numbers all perfectly aligned. One of the prime considerations in choosing a spreadsheet is its ability to generate these convincing lists. Can you modify the format of the printout? A column eight characters wide may be effective on a 40- or 80-column screen, but you may want it to print out as a ten-character column. Remember, your printer is most likely to be 80 columns wide. You are not going to get it to print 100 characters if it is an 80-column printer. Some spreadsheets allow you to copy what you see on your screen directly to your printer as a sort of snapshot of your screen. Others allow you to name the first and last cell that you want printed. Some allow both. There are programs that create graphs from lists of numbers in your spreadsheet. Will the program actually copy the graph to a printer? It is possible if you have the Imagewriter printer; unlikely if you have anything else.

In some business applications it is nice to have a page of a spreadsheet located within the body of a report or letter. This is not as easy as it sounds. You must have a word processing program that works with your spreadsheet.

COMMUNICATIONS PROGRAMS

Telecomputing is simply the process of your Apple //c communicating with another computer through telephone lines. The process is actually rather simple, requiring a special device called a modem that connects a computer with a telephone, and some special communications software. Often a communications program is sold with the modem. You will find a buyer's guide to modems in Part III of this book. A modem and a communications program will open up a whole new world of opportunities.

There are commercial systems involving very sophisticated computers that allow you to connect to and obtain specialized information. One system from Dow Jones, Inc. allows you to obtain stock market information. Key in a stock symbol on your Apple //c and receive up-to-the-minute financial news on that stock, current financial disclosure on the company, and current trading prices. Another system, The Source, allows you to search through catalogs that appear on your screen and actually select and order products from the catalog. Dialog, a California-based system, offers over seven million articles, reports, and books on virtually any technical topic. Of course you are going to have to pay for this information. Generally there is an initial fee when you set up an account with an information service. Then you will have to pay for the time you are actually connected with the service. It can range from five cents a minute to 25 dollars a minute, depending on the type of information and the time of day you access the information.

There are over 2,000 commercially available information services or data banks in this country. These services have some unique advantages over more traditional methods of researching information. First, they are extremely convenient, allowing you to access information from several sources from your own home. Second, they can save you the expense of time and travel usually associated with obtaining information. And finally, the information provided is generally more current than that found in printed works. An excellent guide to available commercial service is *OMNI Online Database Directory* by Mike Edelhart and Owen Davies, Omni Publications International, Ltd., 1983.

Another application within the field of communications is electronic bulletin boards. Users of this service can connect to it just as

they would to a database. They can then enter their own messages, advertisements, or questions from their computer to be viewed by any other users of the system. Electronic bulletin boards are generally for a particular area and they are often centered around a particular interest. There are bulletin boards for Apple users where you can post questions about your computer. There are singles bulletin boards where you can find someone with interests similar to your own. Usually these bulletin boards are inexpensive or free to join.

One application in communications that is finding considerable use in business is electronic mail. Anyone who has tried to contact a busy executive or someone doing field work knows the frustration involved in transferring information. They are out when you call and when they return the call you are out. With an electronic mail service, you enter a message from your computer through the phone lines to someone else with a similar service. The other person can periodically check his or her electronic mail and return the call to your service leaving a similar message. It is possible to have one service that allows many people access to the same information. In this way information can be sent out to many people simultaneously. You could use a commercial service such as The Source to implement electronic mail, or use a Bulletin Board System from public domain. Commercial packages are also available.

All of these applications require some specialized software. A program must be running in your Apple that will allow you to input information from your keyboard and send it through the modem to the receiving system. The same program must be able to receive information from a distant system and display it on your screen. The program must be compatible with the modem you are using, and versatile enough to conform to standards of various data banks. For some applications you may want more from your software. If you want to be able to save information from another system for later viewing, printing, and program execution, you are going to need some additional features. Electronic mail will require some additional software. Communications software will generally work with a variety of modems. The Hayes Smartmodem has become almost a standard for personal computer communications programs, although the simpler programs should run nearly any personal computer modem.

Criteria for Evaluating Communications Programs

Configurability

If you have tried to purchase peripherals compatible with your computer, you have already learned that there are

about as many different types of connections as there are devices. There are parallel and serial connections, IEEE and RS-232 connectors. And, devices used together must share the same handshaking protocol. Configurability refers to the flexibility of your software in making two systems or devices compatible. Your first concern is whether your software will actually work with your modem.

When you attempt to access information from a database or bulletin board, you must first make your computer compatible with another computer. Fortunately, some standards have been established, and any commercial system has gone to great efforts to make it easy for paying customers to connect their system. But there are some parameters that must be agreed upon between two computers before they can interact.

One parameter is the rate at which information is going to be sent. The baud rate refers to the number of bits per second that are being transmitted to or from a system. A standard rate for personal computers connected to most data banks is 300 baud. But some modems and systems allow for a 1,200 baud rate. You will need to be able to enter that information through your communications software prior to establishing contact with a system.

Some systems have an error detection system built into the computer and/or a modem called "parity checking." The computer counts the first seven bits of a byte of information and determines if it is even or odd. If the system uses an even parity detection system it will change the eighth bit to whatever is necessary to make the byte even. When a receiving system checks to make sure that the information sent was not accidentally changed in transmission, it checks to see if the byte is still even. If an error occurred, the receiving system will ask the sending system to repeat the transmission. Odd parity detection systems set the eighth bit to make the byte odd. Two systems must first agree on which, if any, error detection system is going to be used, and your communications software must be able to alter the parity.

Some systems insert an extra bit or two before or after bytes so that the receiving system knows where a character begins and ends. These start and stop bits must be agreed upon and entered through your software. Different computers may use a different number of bits to represent a character. Seven or eight are the most common. When you send information to a receiving computer, it will often return the same character to verify that it has received it. This mode of operation is called full duplex mode. In half duplex mode, the characters will not be echoed back. Again you will want to set this information in advance through your software. Some software will also interact with your disk drive or printer. If this is the case, we will discuss the flexibility of configuration in this section as well.

Special Features With a bare bones communications package you can configure your computer to be compatible with another computer, you can send information from your keyboard to another computer, and you can receive across your screen information from another computer. There are a lot of other desirable features incorporated in some programs.

Information coming from another computer will fill your screen and then scroll out of sight along the top. It would be nice to be able to take that information as it comes in and save it for viewing at a later time. This capability is referred to as capture capability. Usually the information is saved in a buffer in the memory of the Apple. The size of the buffer will limit the amount of information that can be saved at one time. It takes about a 4K buffer to save one page of print. Once you have received the information you may want to save it to disk, send it to a printing device, or view it immediately.

Normally you will send information to a receiving device by entering it from the keyboard. For your application you may be interested in sending an entire file of information previously entered. This uploading capability is found on some software.

Most databases require an initial sequence of entries to establish communications. In this logging-on phase you may be prompted from the database with a question as to your account number. You would enter your account number and then be prompted with a question requesting your password. This entry is a secret entry known only to you and the other computer that prevents other people from using your account.

After entering your password you could be prompted with a question as to the specific information you are looking for. A handy feature in some programs allows you to press a single key and have all of the log-on information sent automatically. In some programs you can save all the configuration information recorded automatically rather than having to re-enter it each time you establish contact.

INTEGRATED SOFTWARE SYSTEMS

The trend in high-quality software is definitely toward integrated systems. These major software packages provide at least two of the traditional major applications: word processing, spreadsheet, database, and communications. Some provide all four features. The big advantage of integrated systems over individual packages is the ability to easily move data from one of the applications to another. A

typical example would be producing a report that combines new text written with the word processor with tables produced by the spreadsheet, and listings or reports produced from a database. It is also handy to be able to easily combine fragments of reports or databases produced at remote locations through telecommunications, and even to send the finished report to another office or company instantly through the telecommunications feature.

While it is not necessarily impossible to do this with a mixed bag of programs, at best it can be difficult and time-consuming, and at worst actually impossible outside of re-keying pages of data. The integrated software system is intended to make these movements of data between applications as simple and fast as possible, and many have succeeded admirably.

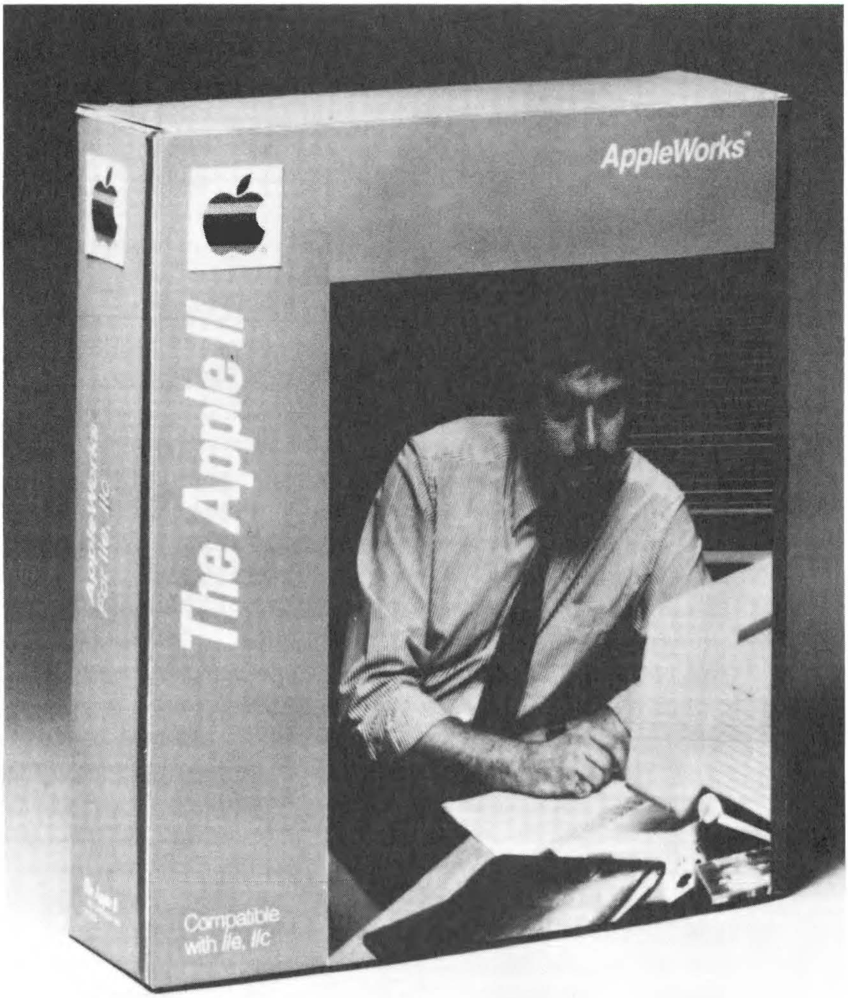
PUBLIC DOMAIN SOFTWARE—WHAT IS IT?

Very simply put, public domain software consists of all the programs that have been intentionally released by the author to the public without copyright notice. These programs may be copied freely and used without restriction or fee. It is literally free software. Thousands of useful Apple programs now exist in the public domain.

With few exceptions, these will run on the Apple //c even though most were developed orginally for the Apple II+, Apple IIe, Apple III (in emulation mode), and the Franklin. Many of the programs are very useful, and are available only in the public domain. Many items of "public domain software" are not programs, but rather templates for spreadsheets, short "fixes" to the Apple software itself or commercial software that does not run as advertised, or other useful non-program material such as programming tips, helpful hints, hardware and software reviews, and tutorials. Others are programs that are less powerful than commercially available programs, but are attractive because they are free and often easier to use. Still others are just good clean (and cheap) fun!

This discussion is a brief overview about public domain software. For a very thorough and detailed treatment of public domain software for the entire Apple family of computers, see *Free Programs for your Apple* by Gary Phillips and Thomas McCord (Sybex, 1984). This book reviews hundreds of programs currently available for the Apple. Directions and hints for using the programs are often included, as well as notes on particular problems that we have

encountered in using some of the programs. One chapter provides pointers on how to cure some common problems with public domain programs. The appendices contain listings of sources of public domain software, user groups, bulletin boards, a discussion of the copyright issue, ordering information, a glossary, and an index to the book.



Appleworks is Apple's integrated software system for the Apple //c. It offers integrated database management, word processing, and spreadsheet analysis.

Is it Worth the Effort?

You can pay $50 for a program to handle your mailing list, or you can get one free from the "public domain" through a local computer club. There are dozens of professional consultants who will gladly design a bookkeeping system for you, and you will pay dearly for it. Alternatively, you can use a system that someone halfway across the continent designed–a system that is available to you without charge as public domain software. Whether you want a new game, a business application program, a household money management system, or even a computer programming language, there is a fair chance that a free public domain program exists to fill your needs. Often, the program you want may be available only in the public domain, or the public domain version may be superior to anything available commercially.

If you are willing to accept the program without the "bells and whistles" and all the fancy options, you may be able to save hundreds of dollars on software purchases. In some cases you will be surprised to discover that your free program serves you better than the expensive program purchased by a friend.

Who Writes It?

A network of home computer users has grown in this country, in fact worldwide, over the last decade. Meeting in homes and clubrooms from Boston to Kansas City to Los Angeles and all points in between and beyond, these hobbyists help one another with their problems in learning a new technology. They share successes and failures with computers and programs. When one person comes up with a problem, someone else is usually there to suggest a solution. As this self- help network has grown, people have begun to share not only their experience and their information, but also the computer programs they have written for their own use.

Working together, they have produced a vast library of public domain software–computer programs that have been donated for the public good so that anyone may use them as he or she pleases. All of the programs are listable so you may tailor them to your particular application or system. Best of all, public domain software comes to you on an unprotected disk. You do not have the headaches and moral dilemmas associated with copying protected disks.

Some public domain authors are established computer professionals who have their careers well in hand. They have no interest in selling programs. Many of them work on large mainframe systems

and tinker around with a home computer. Others are professional writers, doctors, teachers, insurance salesmen, high school students, and even grade school students.

A great majority of the programs are just short utilities written to rectify a particular problem that the author encountered, solved, and he or she wishes to share the solution with others. Many just display a new effect or feature of the Apple discovered by the programmer. But a surprising amount of the software are well-planned out and executed programs or series of programs. These run the gamut from simple yet clever math drills to games of almost arcade quality to sophisticated accounting packages.

You can become part of this network. By contacting the sources listed in the appendices of this book, joining a user group or making a phone call using your computer and a modem, you can have free access to programs that would cost you hundreds, perhaps thousands, if purchased commercially.

Is it Really Free?

The software itself is absolutely free. Any expense incurred will be a small handling charge for the cost of the disks, postage, and labor. The fees charged range from a few organizations that will allow you to copy their software for free to commercial concerns that charge you $15 a disk and up. However, one must remember that this cost is not for one program but for a disk full of programs.

Many user groups allow you to copy computer programs on your own disks at their meetings, or the club will sell you the disk at a nominal cost at their meetings or through the mail. Still others maintain electronic bulletin board systems (BBS), where you can call up with your computer and load a computer program without ever having to leave your home. There are large computer time-sharing services, such as The Source and Compuserve, that maintain large libraries of public domain software (a service often financed, at least in part, by computer manufacturers).

Indeed, many computer groups have sprung up that concern themselves with a very specialized application. This can vary from users that concentrate on hardware, professions such as dentistry or agriculture, and programming languages such as Logo and Pascal to those that are mainly preoccupied with a particular type of software such as education, business, games, or graphics. Some groups have even narrowed down their area of interest to a single piece of software such as dBASE II, VisiCalc, or Wizardry. Most computer clubs have special interest groups

that discuss particular applications in addition to their general meetings.

That Stuff Must Be Junk!

Even though there seems to be a great deal of advantage to obtaining this "free" software, perhaps some of you are still put off from rushing out to acquire some because of the concept that anything is worth exactly what you pay for it. Admittedly, there are a lot of public domain programs that should never have seen the light of day. The same, unfortunately, may also be said for some high-priced commercial programs. We all have a box or more full of rarely-used commercial programs that turned out to be major disappointments. It is far more aggravating and expensive to make a wrong decision on a $40 program than a program that cost you 40 cents.

Happily, for all the uninspiring programs you may pick up in the process, there are great numbers of public domain programs that will make the search for good, inexpensive software pay great dividends. The joy of public domain software lies in finding one of these "diamonds in the rough."

User Groups

Going to a meeting of a local user group is an excellent way to obtain public domain programs. Since the Apple //c is highly compatible with the earlier members of the Apple II family, most Apple user groups will include users of all the Apple computers. Some may have special interest groups for the //c, and no doubt a few organizations will spring up that are devoted exclusively to the //c because of its unique advantages of portability and its closed-system architecture. Most clubs maintain a library of public domain programs that have either been written by the members or obtained from other computer clubs. Some smaller groups will allow you to make copies before or after the meeting, so the copies are truly free (except for your time and your disk). Some even put together a "Disk of the Month" that they provide to the membership for free or at a minimal cost. This disk will either be copied onto your own disk or you can buy a disk that has been prepared in advance. The price of this disk will vary depending on the groups' idea of the cost of "handling," varying disk prices, and whether or not the club uses public domain program sales as a way to help finance its operations. Appendix B lists user groups worldwide.

Mail Order Sources

Many clubs and organizations will mail copies of the programs in their collections to you. Some will copy the programs onto the blank disks you send them, while others prefer to provide the programs on their own disks. Of course, there will be some added postage and handling fees but you will have a much greater number of programs to select from than you could possibly find at the local computer club.

A somewhat more involved, but very workable, way to get public domain programs is through telecommunication. A number of individuals, clubs, and commercial organizations maintain collections of public domain software compatible with the Apple //c that you can access with a modem and a communications program. You can copy these programs from the remote database onto your own disk with no charge except phone charges (and for a few commercial services a connect time charge).

For Apple //c owners who are not members of active user groups, and for many who are, bulletin boards provide a major source of public domain software. The other major sources of free software by telephone are the commercial bulletin board systems, The Source and Compuserve.

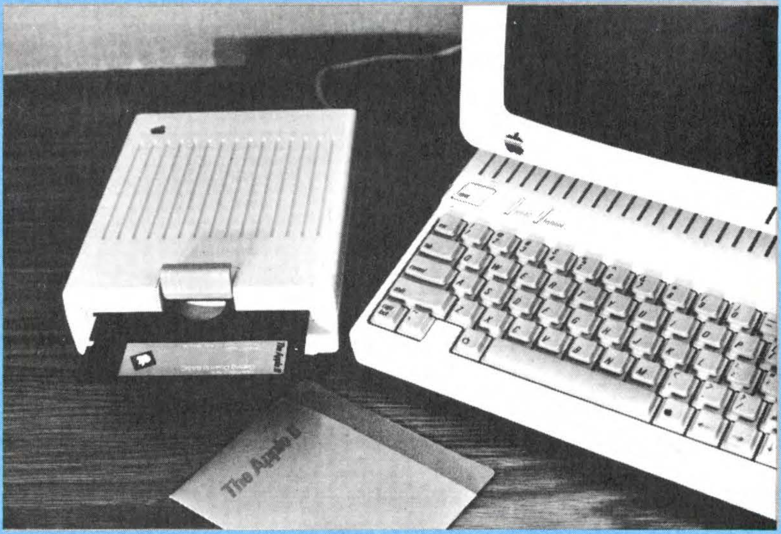
Both The Source and Compuserve are miles away from the simple bulletin board systems run by so many amateurs across the country. They provide hundreds of services besides the bulletin boards—everything from weather reports to stock quotations to encyclopedias to computer games to advice for the lovelorn. For a fee, they both allow home computer owners to call and upload or download public domain programs.

To make use of any of these services, bulletin boards or commercial services, you must have a modem. For guidance on modems, see Chapter 15.

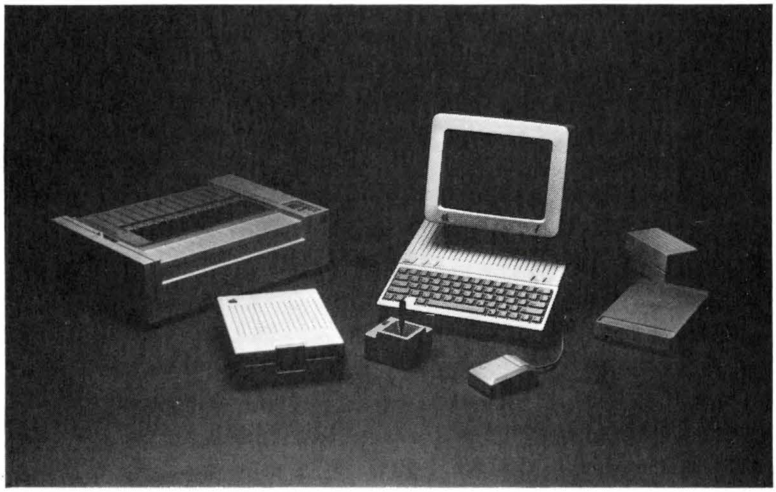
Apple //c Hardware Expansion Buyer's Guide

11

How to Select Expansion Hardware for Your Apple //c



The Apple //c is probably the most useful device you own. What sets it apart from most other appliances and machines found in the home is its versatility. The same computer that will alphabetically sort a list of names, edit a document, and calculate the mortgage will also generate a graphic display, challenge you in chess, and teach you to type. While the Apple //c is extraordinarily well-equipped with built-in equipment such as a disk drive, 128K of RAM, and plug-in sockets (or "ports") for a monitor and printer, there are many accessories you will need for certain functions. A color monitor adds tremendously to the appeal of graphics and games, while a second disk drive or hard disk facilitates editing large documents with your word processing program or keeping track of large files with your database program.



Apple offers a full range of peripherals for the Apple //c. Pictured here (from left to right) are: the Imagewriter printer, Disk //c external disk drive, carrying case, joystick, Monitor //c, AppleMouse, and the Apple Modem 1200. The device behind the Apple Modem 1200 is the external power supply provided with the Apple //c.

This guide to Apple //c expansion hardware will give you the necessary background to understand the logic and terminology of expansion devices, what the various types of hardware are, and what they are good for. In addition, we will mention the characteristics of a number of devices that we have used that you may want to consider buying for your Apple //c.

Among the external devices, called *peripherals*, which you might add to your Apple //c are printers, monitors, disk drives, modems,

joysticks, and a mouse. Each peripheral could have its own unique method for transferring data. Some peripherals are indeed designed to work with a single type of computer. But the majority of peripherals are designed to implement a standard used by a variety of computers. This creates a larger market for the peripheral vendor and increases the number of peripherals compatible with a particular computer. It also creates considerable confusion for the user attempting to select a peripheral.

The Apple //c has six ports or sockets to connect various peripherals. These ports are identified by icons, (see Chapter 3). While most of these ports comply to the commonly accepted standards used by most peripherals (such as RS-232 for the two serial ports), the connectors are special to the Apple //c. Most expansion products for the Apple //e and Apple II are electrically compatible with the //c, but the cables supplied will not fit the ports on the back of the Apple //c. New devices designed specifically for the //c will have the correct cables. We also expect to see adapter cables available to allow Apple //e and Apple II hardware to run on the Apple //c. These may be produced by Apple, by independent companies, or both. The connectors of some Macintosh and Lisa peripherals such as the Mouse for Mac use the same connectors as the //c and will work on the //c. The Imagewriter printer, often used with Mac, will work on the Apple //c with a special cable. In terms of connectors, the Apple //c is consistent with the Macintosh and Lisa rather than the earlier Apple computers.

If you wish to go further in your understanding of how a peripheral works, we have included an introductory section in each chapter to explain how each device works and what capabilities it can add to your Apple //c. We hope that the introductions and brief descriptions of products will get you off to a good start in understanding expansion products for the Apple //c, and in making wise choices from among the many available devices.

BITS AND BYTES

If you do not really care about how things work inside the Apple//c, feel free to skip to the "Parallel and Serial" section below. But if you want to look under the hood of your Apple //c, here we go!

Information in any computer is represented and stored as voltages. Imagine having eight wires protruding from a black box. The black box can generate voltages on any combination of the eight wires. You could assign and represent a character in the alphabet with a particular combination of voltages. You could let the letter A

be represented by applying voltages on the first wire only. Letter B could be represented by applying voltages to the second wire. Letter C could be represented by applying voltages to both the first and second and so on. This is exactly how computers store information. Microscopic circuits store voltages that represent characters, numbers, and punctuation marks. You can represent 256 different characters without repeating the same combination twice using only eight wires. This is more than enough to represent all of the characters in the alphabet, the numbers zero through nine, and most punctuation marks. To help remember the combinations of voltages that represent each character a numbering scheme has been developed.

If you laid the eight wires out flat and placed a one on each that had a voltage applied and a zero on the ones that did not have a voltage, the number 00000001 could be used to represent the letter A described above. The number 00000010 would represent B, and 00000011 would represent C. This numbering scheme is called binary since combinations of two numbers are used to represent all the values. An individual voltage, called a bit, for binary digit, and the group of eight voltages representing a character is called a *byte*. These binary numbers have decimal equivalents. Without explaining the conversion process, the number 1 would be equal to 00000001, the number 2 would equal 00000010, and the number 3 would equal 00000011. Using this decimal scheme, the letter A could be represented by the number 1, the letter B by the number 2, and the letter C by the number 3. Essentially we have created a code to represent all the characters we wish to store in the computer.

A number of codes have been developed for use in computers, but by far the most common one is *ASCII*—American Standard Code for Information Interchange. The important fact to remember is that characters are stored as voltages and represented by numbers. If you typed the word HOUSE on the keyboard of your computer and had some microscopic probes, you could actually find the voltages representing each letter of the word HOUSE somewhere on one of the chips in your computer. The eight voltages collectively are referred to as a byte of data and an individual voltage is referred to as a bit.

MOVING BITS AND BYTES

The computer has only these bits and bytes to work with. It can add them, subtract them, compare them, and move them around.

You can attach independent devices called peripherals and move the bytes of data to them. A printer will take the voltages that represent the letter A and automatically print the letter A on paper. A disk drive attached to the computer will take the same pattern of voltages and magnetize a diskette to permanently store the letter A. A modem will take the voltages representing A and translate them into audio signals that are sent across a phone line and converted back into digital voltages.

It would have been nice if all manufacturers of peripherals and computers had gotten together before developing all these machines and agreed upon a single method of transferring data. You could have purchased any peripheral and expected it to just plug in and start working with your computer. In reality very few devices just plug into a computer and start working. Devices maufactured exclusively for the Apple //c or definitely labeled as compatible with the Apple //c should give you a minimum of problems. For more general devices, which include some of the finest products available and much of the "bargain" hardware, it is going to require some knowledge and effort on your part to obtain a peripheral that will effectively work with the Apple //c.

PARALLEL AND SERIAL

There are two general approaches to transferring information between computers and peripherals. *Parallel transmission* transfers a complete byte, eight bits, simultaneously. At least eight separate wires or lines are needed and usually many more are used.

An alternative to sending all eight bits simultaneously is to transmit *serially* one bit at a time. The transmitting device breaks a byte of data into eight separate bits and transmits each, one at a time. The receiving device collects each bit and regroups it into the original byte. This serial transmission of data is usually slower than parallel transmission but requires only one line for the connection.

There are two approaches to sending data serially. If the transmitting and receiving devices have been set to work at a common speed, and if special start and stop bits are sent to indicate when one entire byte has been sent and another is about to be sent, the transmission is called *asynchronous transmission*. If a chain of bytes are transmitted without start and stop bits, the transmission is synchronous. With synchronous transmission there may be some preliminary data sent to the receiving device to let it know when transmission is to begin and how much data is going to be sent.

Many devices are available with either a serial or a parallel interface. Devices made for the Apple //c will usually have an RS-232 serial interface or an interface consistent with one of the other built-in ports (disk port, mouse port, and so forth). Parallel devices will require a serial-to-parallel adapter such as the Discwasher adapter.

HANDSHAKING

It seems easy to transfer voltages representing a character from one location to another. Simply connect wires from the source location to the destination. When more than one byte of data is to be transferred, however, the receiving device must know when the first byte has been received and the second byte begins. It must also know when the last piece of data has been received. The computer, likewise, must know when the receiving device is ready to accept a second piece of data and when the last piece has been accepted. This process of sending additional information that controls the flow of data between computer and peripheral is termed *handshaking*. The format for handshaking (the actual sequence of events as information is traded back and forth) is called *protocol*. There could have been countless possible ways of transmitting data between computer and peripheral. The computer engineers could have designed an infinite assortment of cables and plugs to make the physical connections and infinite protocols to transfer data. Fortunately, they held off at about three or four standards. They left enough variations within those standards to confuse things but still, all in all, we got off lucky.

The three most common standards used among personal computers and peripherals are RS-232, Centronics Parallel, and IEEE. Apple //c uses can only use RS-232 easily, but it is good to know a little about the others. For Centronics Parallel devices, you will need an adapter such as the Discwasher serial-to-parallel adapter.

STANDARD RS-232

One common standard of data communications is RS-232. Used in many modem connections and printer connections, this standard specifies the voltage levels and the required driver and receiver characteristics for 21 circuits defined by the standard. Information is sent one bit at a time in a serial fashion between devices. Sometimes serial transmission is used loosely to mean RS-232 transmission.

The physical connections are defined with 25 individual wires, each having a prescribed use. However, vendors of equipment using RS-232 standards are in no way compelled to use all of the circuits described. Many applications for personal computers use only three of the wires defined. Although the most common plug used is the DB-25, vendors often vary from that standard as well. In addition to the variations of plugs and cables there are additional parameters that are left undefined.

Some of those parameters such as the baud rate, which is the rate that data is transmitted, are described in greater detail in Chapter 15 on modems. But the bottom line on RS-232 is that there are plenty of variations within the standard to cause trouble when connecting devices. The Apple //c serial ports both use a five-pin DIN connector, and will require an adapter or special cable for devices using a DB-25 or Apple //e card connector.

EXPANSION PRODUCTS GLOSSARY

The following terms occur frequently in discussions of peripherals that you might wish to attach to your Apple //c.

alphanumeric A letter of the alphabet, a numerical symbol, or any other set of characters is called alphanumeric.

ASCII American Standard Code For Information Interchange. Information is stored in the computer using different combinations of eight voltages to represent different alphanumeric characters. ASCII is a common standard that defines which combinations of voltages represent which characters.

asynchronous Two general methods of serial transmission are synchronous and asynchronous transmission. In asynchronous transmission a receiving device is informed of the beginning and end of each character with additional bits (start and stop bits).

baud rate This term is essentially a synonym for the transmission rate in bits per second. A 300 baud modem can transfer 300 bits per second, or about 30 characters per second.

bit Characters are stored in computers as combinations of eight voltages. An individual voltage from the group of eight is referred to as a bit.

buffer A temporary storage area for data.

byte Information is stored within a computer using combinations of eight voltages to represent different alphanumeric characters. The entire set of eight voltages is referred to as a byte.

Centronics Centronics Parallel is a standard for transferring data between two devices. Used by many printers, the standard defines the handshaking protocol used in the transfer. The Discwasher serial-to-parallel adapter provides a Centronics Parallel port for the Apple //c.

duplex This refers to the ability to simultaneously transmit and receive data over a single line. Most often associated with RS-232 transmission, full duplex capabilities are implemented by using two different frequencies for transmitting and receiving on the same phone line.

echoplex With some transmissions data is transmitted to a remote computer and returned to be displayed on the originator's video display unit. This process verifies that the transmission was received correctly.

handshaking When data is transferred between two devices, additional information controlling the flow of data is required. This additional information usually involves signals sent back and forth between devices and is termed handshaking.

IEEE This refers to a standard for transferring data between devices. It defines the handshaking protocol used in the exchange. IEEE involves the parallel transfer of data and allows for multiple devices to be attached to the same bus.

input device A peripheral device used to enter information into a computer is referred to as an input device. A keyboard and graphics tablet are both input devices.

output device A peripheral device used to display or reveal information from a computer is termed an output device. Printers and monitors are output devices.

parallel Information is stored within a computer using combinations of eight voltages to represent characters. If all eight voltages are transferred to another component or device simultaneously the transfer is called a parallel transmission.

parity Information transmitted over large distances is subject to interference and distortion, which may result in error. Parity is a scheme devised to check for errors brought about during transmission. With an even parity check a transmitting device counts the number of "on" bits in the first seven bits of a word. The device toggles the eighth bit to whatever is necessary to make the sum of eight bits even. The receiving device totals the eight bits and verifies that the sum is indeed even. If not, an error occurred with a single bit during transmission and the receiving device requests

that the word be repeated. If two bits were subject to error, parity will not detect the error.

peripherals Devices connected to and used in conjunction with a computer are called peripherals. Included are printers, modems, disk drives, and mouses.

protocol When data is transferred between two devices additional information must be included to indicate when a receiving device is ready, when data has been accepted, and where data begins and ends. The format and procedure for exchanging this information is called protocol.

RS-232 A standard used by most modems and many printers for transmitting information. Data is sent serially one bit at a time. The term serial transmission is sometimes used loosely to refer to the RS-232 standard. Serial ports 1 and 2 of the Apple //c, intended for the printer and modem, respectively, are both RS-232 ports and should support most standard RS-232 devices.

serial Information is stored in a computer as voltages within specialized circuits. Alphanumeric characters are represented by combinations of eight voltages. A serial transmission of a character involves sending each of the eight voltages one at a time.

start bit When information is transferred serially in an asynchronous method, additional bits marking the beginning and end of a character are also transmitted. These extra "framing bits" are called start and stop bits.

stop bit See start bit.

synchronous Two general methods of serial transmission are synchronous and asynchronous. Synchronous transmission sends information as groups of characters. The beginning, end, and length of the group is usually included in the transmission.

word Information is stored as voltages within specialized circuits of a computer. In personal computers, eight voltages or bits (one byte) are usually used to represent a character. A word describes the largest group of bits handled as a single unit by the central processor. In the Apple //c a word is 16 bits (two bytes).

12

Monitors

THE MONITOR, CRT, OR SCREEN

The Apple //c must have a visual output device to communicate with you. You have many options for this device: the special portable flat screen designed for the Apple //c, the standard monitor with its special stand for the //c, various monitors in green, black and white or amber, color monitors, and your household TV set (whether black and white or color). Any of these will function quite well with the Apple //c. Some are better at particular functions than others, and there are significant price differences.

Just a year ago a monitor was a significant additional expense for the personal computer purchaser. Now that mass production of monitors has brought prices down, a good quality 12-inch monitor can be purchased for as little as $60 from mail order or discount computer stores. This makes sharing the household TV with the computer less of a necessity for everyday computer use. The 40-column mode of the Apple //c and many other computers is specifically intended to allow the computer to generate letters large enough to be legible even on a slightly fuzzy TV screen. On the crisper, higher-resolution screen of a monitor, 80 columns or more are easily legible.

But the TV may well still have a place beside your Apple //c, for it is still a very practical way to enjoy the remarkable color capability of the Apple //c. Color monitors are quite expensive, and a good color TV will do a fine job of displaying games and other graphics. A color monitor will do an even better job, and the interface for a composite color monitor is built into the Apple //c. RGB(red-green-blue) color monitors may also be used with an adapter such as the Video 7 RGB adapter.

How do monitors and TVs work? Both use a CRT (cathode ray tube) to generate images. We used to call this a picture tube back when TV repairmen made house calls. It is the primary component in any TV or monitor. The CRT is a glass tube that contains a filament that when heated to the right temperature emits electrons. These electrons are focused into a beam and directed with magnetic devices within the tube. The beam is aimed at a phosphorus screen located at one end of the tube. Phosphors have the unusual property of momentarily glowing when struck by electrons. When we view the screen from outside the tube, sections that are being struck by the electron beam will appear illuminated.

Starting at the upper left corner of the screen the beam is directed from left to right at a slight downward angle. When it arrives at the right edge of the screen it shuts off and moves to the left edge of the screen. The beam continues back and forth descending slightly each time until it arrives at the lower right corner. From there it jumps

back to the upper left corner and repeats the process. In this way the beam manages to scan the entire surface of the screen. The trick in getting the CRT to display an image is to control the intensity of the beam as it makes its way across the screen. By turning the beam on or off at the right places we can illuminate various sections of the screen to create an image. If we slowed the process down we would see each illuminated section glow for a moment and then fade out. But since the beam is actually scanning the screen 60 times a second (faster than our eyes can perceive the changes), the illuminated sections appear to remain on continuously.

Color TVs and color monitors work in a similar fashion. Instead of one electron beam, three beams—red, blue and green—are generated. Instead of a single type of phosphor, three types cover the screen. Each of the three beams is focused onto a point of the appropriate color. By varying the combinations and intensities of the three beams different shades and colors can be generated. Again the trick in getting a color display is to get the right beams to come on and off at the right place on the screen.

THE MONITOR OR TV

What is the difference between a monitor and a TV? When we enjoy a TV show the information that determines the images of the show we are watching are transmitted by high frequency radio waves from a TV transmitting station. Special circuitry in our TV isolates the particular channel signal from all the others and filters out the information for the screen. If we want to connect our computer to a TV we must produce a signal for the TV similar to the type the TV would get from an antenna. To do that we modulate the information. That is, we make it appear as if it is coming in from channel 3. This is done by a special device called a Radio Frequency Modulator (or RF Modulator). This small device is included with the Apple //c, along with the adapter to connect it to your TV and a cable to connect the two together. The RF modulator is marked with an icon that matches the color connector on the back of the Apple //c (the same connector that would be used to attach a color monitor). The TV picks up the modulated signal as if it were coming from channel 3 and goes through its normal operation of demodulating the signal and filtering out the information that tells it when to turn the electron beam on and off. A monitor does not have the circuitry to isolate a TV channel and filter out information. It is not necessary to first modulate the signal and then demodulate it.

The advantage of avoiding all that modulation and demodulation appears on the display. The resolution and clarity of a monitor is

noticeably better than a TV. In addition, you avoid a lot of the interference that may develop on a TV. Remember that your computer's signal is sent as if it is on channel 3 or 4. The real channels 3 and 4 are also transmitting signals and they may also appear on your screen.

SELECTING A MONITOR

Almost any monitor is going to improve the image clarity if you are used to a television as a display unit. But sorting through the many monitors available to obtain the best for your needs is a little more difficult. If you could put two monitors beside each other and compare the pictures it would help. But few retailers are that accommodating. A few simple tricks might help you along. Fill the screen with a single character, preferably in white. Step back a few feet and see if the characters are aligned properly. That is, make sure that the rows are horizontal and columns align vertically. Make sure that the characters are as clear along the edges of the screen as they are in the middle. With a color monitor look for colored fringes on characters around the edges of the screen. This may be an indication of poor construction.

Make certain the intensities of the characters are even, and that characters along the edges are as bright as those in the middle. Many monitors have an assortment of control knobs including those that control the brightness and contrast. Some allow you to better align the image horizontally and vertically. As soon as the salesperson is gone play with the knobs to get a feel for the limitations of the monitor.

Hold a cursor key down to move it at a maximum speed across the screen. Is there a trail behind it indicating that the persistence of phosphor is too long? This will become very annoying when working with text that scrolls up or down and leaves momentary ghosts of previous characters.

MONITOR SPECIFICATIONS

There are many factors affecting the quality of a monitor. The only important one is the quality of the image produced on the screen with your computer. Technical specifications, however, provide a starting point in selecting a monitor. They give you an insight as to which monitors should produce the higher quality image. In some cases, where it is impossible to test your Apple //c with a monitor,

the specifications may provide the only assurance that your monitor is going to work adequately with your computer.

The first thing to consider in selecting a monitor is whether you want and can afford a color monitor, or whether your application only requires a monochrome monitor, a single color display. Monochrome monitors are occasionally in black and white but more often either green or amber. The contrast of black and white can be hard on the eyes after extended viewing. Green seems easier on the eyes, and some argue that amber is easier yet. It is best to take a look at both before you decide.

There can be a problem with monochrome monitors. Many of the programs for the Apple //c are displayed in color. Some colors that appear differently on a color display are identical on a monochrome display. Be aware of this potential problem, and if there is a single application you are concerned with make certain that there is not a display problem on a monochrome screen before you buy.

If you are going all out and getting a color display there are two types: RGB and composite video. The Apple //c can use either, although an adapter such as the Video 7 RGB adapter is required for RGB monitors. With RGB units the information that turns the three electron beams on and off is sent through on individual lines. With composite video the information for color is sent through a single cable. Another specification concerns the screen resolution. Many manufacturers indicate the number of pixels that can appear on the screen. The greater the number of pixels the sharper the image. For its double hi-res mode, the Apple //c needs a monitor capable of displaying 560 × 192 pixels.

Color monitors direct their three beams through a perforated screen called a shadow mask just before they strike the phosphorus screen. This shadow mask helps converge the beams. The dot pitch indicates the distance between the holes in the shadow mask. The smaller the distance the better the resolution.

The Apple //c can connect the computer to each of these devices as follows: RGB monitors are connected to the computer via the video port on the back panel of the computer with the Video 7 or similar adapter; color TVs, black and white TVs, and hi-res monitors are connected via the video jack in the center of the computer's back panel.

With all TVs and monitors there is a limit as to how much information they can accept from the computer each second. Monitors with greater bandwidths will display sharper images. Greater bandwidths generally mean better pictures.

Another specification that affects the display is the phosphor persistence. This is an indication of how long the pixels remain visible after being struck by the electron beam. Screens with too short a

persistence will appear to flicker. On the other hand, those with a peristence that is too long will have a problem with animated displays. Text as it scrolls up on the screen will leave an annoying ghost image on screens with a phosphor persistence that is too long.

INTERFACE

Most monitors utilize a special cable and plug designed to reduce interference. It is usually an audio cable terminated with RCA plugs. RGB monitors ususally use a special connector and require the Video 7 or similar adapter. The nice thing about monitors is their compatibility with software. Although the quality of the picture may vary with monitors, as long as you are able to view the screen from one program you can rest assured that you will be able to view the screen with any other program.

SUMMARY

Monitors and televisions start out with many things in common. Their principle component is a Cathode Ray Tube (CRT). This device generates an electron beam that is directed against a phosphorus screen at the end of the tube. The beam is directed across the screen from left to right, descending slightly as it moves. When it reaches the right side of the screen, it is shut off and moves back to the left side. This process is repeated over and over until the beam finally arrives at the lower right corner. At this point the beam is temporarily turned off while it is redirected at the upper left corner where it begins the entire process over again. In this way a single beam covers the entire screen. By moving the beam fast enough and turning it on and off at appropriate locations an image can be generated on the screen.

Monitors and televisions are similar in purpose and function. In both the primary component is the CRT. A television, however, is designed to receive information on high frequency radio waves. The radio waves must be separated from the information that controls the electron beams of the CRT. To use a television with a computer, the information from the computer must first be modulated to simulate the high frequency transmissions. A monitor can accept the information directly without modulation. By avoiding the modulating and demodulating process, a cleaner, crisper resolution can be produced on the screen. Monitors can produce a finer display resolution resulting in less eye fatigue.

Here is a glossary of terms used in reviews and descriptions of monitors.

bandwidth The bandwidth of a monitor or television is the maximum frequency response of the monitor. The quantity provided by specification sheets is given in megahertz or millions of cycles per second. Higher bandwidth means sharper details.

brightness One factor in the quality of an image depends on the intensity of light generated from an image. Most monitors have a single signal that controls all three electron guns.

contrast One factor in the quality of an image depends on the difference in light intensity of an image and its background. Most monitors have a contrast control.

convergence This refers to the accuracy with which the electron guns hit individual pixels.

CRT(Cathode Ray Tube) The principal component of most televisions and monitors. The CRT generates one or more electron beams at one end of the tube and directs it against a phosphorus screen at the other. By scanning the beam across the entire screen and varying the intensity of the beam images can be created.

dot pitch The distance between the holes in a shadow mask.

electron gun An electron gun is a device inside a CRT that generates beams of electrons used to display an image.

horizontal linearity This refers to the horizontal distortion that may appear on some monitors. If the width of characters on a line varies it may indicate poor horizontal linearity.

monitor A device utilizing a CRT, flat screen LCD, or similar device to display images similar to a television. Whereas a television receives its information on high frequency modulated waves, a monitor receives its information directly from a source such as a computer.

pixel The contraction of picture element. As an electron gun strikes the phosphor screen dots of light called pixels are created.

phosphor persistence When an electron beam strikes the phospor screen of a CRT light is emitted. The amount of time the phosphor continues to emit light after being struck is called the phosphor persistence.

raster scan The electron beam in a CRT is directed back and forth across the screen in a process called raster scan.

refresh rate An electron beam begins scanning a screen in the upper left corner. It moves left to right descending slightly after

each line until it reaches the bottom right corner and the entire screen has been covered. It then jumps back to the upper left corner and repeats the process. The rate at which the scanning is repeated is called the refresh rate.

RF modulator A small device that is used to connect a TV to a computer. It modulates the signal the TV is receiving so that the TV can demodulate the signal just as it would a normal TV station channel.

RGB (Red, Green, Blue) In an RGB monitor there is special circuitry that allows for three separate signals to control the three electron guns. A Video 7 or similar adapter is required to use an RGB monitor on the Apple //c.

screen memory A section of memory in a computer that stores information to determine what will be displayed on the screen.

screen resolution This is a reference to the number of pixels that can be displayed on the screen.

shadow mask A shadow mask is a perforated sheet of metal that lies behind the phosphor screen. The three electron beams pass through the same holes to focus on a single pixel.

video controller A device used to organize and send the information from a computer to a television or monitor.

The following table lists the monitors that are available for use with the Apple //c.

Monochrome Monitors.

Manufacturer	Model	Type	Diagonal Size	Pixel Resolution	Text Resolution
Amdek Corp	V-300	green	12 in.	900 × 800	80 × 25
	V-300A	amber	12 in.	900 × 800	80 × 25
	V-310	green	12 in.	900 × 800	80 × 25
	V-310A	amber	12 in.	900 × 800	80 × 25
	V-310G	green	12 in.	900 × 800	80 × 25
Apple Computer	Monitor II	green	12 in.	560 × 192	80 × 25
	Monitor //c	green	9 in.	560 × 192	80 × 25
	Monitor III	green or B/W	12 in.	480 × 192	80 × 24
Computer Systems	VD/9	green or amber	9 in.	400 × 1024	80 × 25
	VD/12	green or amber	12 in.	400 × 1024	80 × 25

Monochrome (continued).

Manufac-turer	Model	Type	Diagonal Size	Pixel Resolution	Text Resolution
	VD/19	green or amber	19 in.	400 × 1024	80 × 25
Electrohome	ECM 926	green or B/W	9 in.	n/a	80 × 40
	ECM 1226	green or B/W	12 in.	n/a	80 × 40
NEC Home Electronics	JB-1260M(A)	green	12 in.	600 × 230	80 × 25
	JB-1201M(A)	green	12 in.	800 × 250	80 × 25
	JB-902M(A)	green	9 in.	700 × 230	80 × 25
Sanyo Business Systems	DM2012	B/W	12 in.	n/a	64 × 20
	DM2112	green	12 in.	n/a	64 × 20
	VM4509	B/W	9 in.	n/a	80 × 24
	DM5109CX	green	9 in.	n/a	80 × 24
	DM8012CX	B/W	12 in.	n/a	80 × 24
	DM8112CX	green	12 in.	n/a	80 × 24
TSK Electronics (Taxan)	KG-12N	green	12 in.	n/a	80 × 25
	KG-12N-MN	amber	12 in.	n/a	80 × 25
USI International	Pi-1	green	9 in.	560 × 240	80 × 24
	Pi-2	green	12 in.	560 × 240	80 × 24
	Pi-3	amber	12 in.	560 × 240	80 × 24
	Pi-4	amber	9 in.	560 × 240	80 × 24
Zenith Data Systems	ZVM121	green	12 in.	n/a	80 × 25

Color Monitors (Adapter Required).

Manufac-turer	Model	Type	Diagonal Size	Pixel Resolution	Text Resolution
Amdek	Color I	composite color	13 in.	260 × 300	40 × 24
	Color II	RGB	13 in.	560 × 240	80 × 25
	Color II-A	RGB	13 in.	560 × 240	80 × 25
	Color III	RGB	13 in.	260 × 300	40 × 24
	Color IV	RGB	13 in.	720 × 420	96 × 25

Color Monitors (continued).

Manufacturer	Model	Type	Diagonal Size	Pixel Resolution	Text Resolution
Computer Systems	VD/9	composite color	9 in.	400 × 1024	80 × 25
	VD/12	composite color	12 in.	400 × 1024	80 × 25
	VD/19	composite color	19 in.	400 × 1024	80 × 24
Electrohome	ECM 1302-1	RGB	13 in.	370 × 235	64 × 40
	ECM 1302-2	RGB	13 in.	580 × 235	80 × 40
	ECM 1301	RGB	13 in.	720 × 512	80 × 40
NEC Home Electronics	JC1203DH(A)	RGB	12 in.	690 × 280	80 × 25
	JC1202DH(A)	RGB	12 in.	690 × 280	80 × 25
Sanyo Business Systems	DMC 6013	composite color	13 in.	n/a	64 × 24
	DMC 6113	RGB	13 in.	n/a	80 × 24
Texas Instruments	PHA 4100	RGB	10 in.	256 × 192	40 × 24
TSK Electronics (TAXAN)	RGB Vision I	RGB	12 in.	n/a	40/80 × 25
	RGB Vision II	RGB	12 in.	n/a	64/80 × 25
	RGB Vision III	RGB	12 in.	n/a	80 × 25
Zenith Data Systems	ZVM 134	RGB	13 in.	680 × 460	80 × 25

Comments on individual monitors:

Amdek Monitors The following are arranged in order of price:

12-inch B/W model 100. Handy size, good display, and fine for 80-column display.

Video-300 is a 12-inch green phosphor video monitor. Features include a non-glare screen for no-strain viewing, and easy reading and operation; 18 MHz bandwidth and 900 lines (center) of resolution; compatibility with every computer or word processing system; UL, FCC approved; and composite input.

Video-300A is a 12-inch amber phosphor video monitor. Features include non-glare screen for no-strain viewing, and easy reading

and operation; 18 MHz bandwidth and 900 lines (center) of resolution; compatibility with every computer or word processing system; UL and FCC approved, and it features composite input.

Video-310A is a 12-inch amber phosphor video monitor. The system features a non-glare screen for easy reading and operation. It includes an 18 MHz bandwidth and 900 lines (center) of resolution.

Video-310G is a 12-inch green phosphor video monitor. The monitor features a non-glare screen for easy reading and operation. It includes an 18 MHz bandwidth and 900 lines (center) of resolution.

13-inch color model I. Good color, not recommended for 80-column display.

13-inch color model II. Excellent high-resolution, can be used for 80-column display.

Apple Monitor II A high-resolution, green and black video monitor that will display 80 characters per line and 24 lines per screen. The Monitor II has a 12-inch screen that tilts for easy adjustment of the viewing angle. It connects to the Apple //c video display interface on the rear surface of the machine. *Apple Computer, Inc.*

Apple //c Monitor and Stand Designed especially for the Apple //c, this small, streamlined monitor has an optional stand to support it in an optimal viewing position above the Apple //c. The stand also serves to tilt the Apple //c computer at an angle for easy reach.

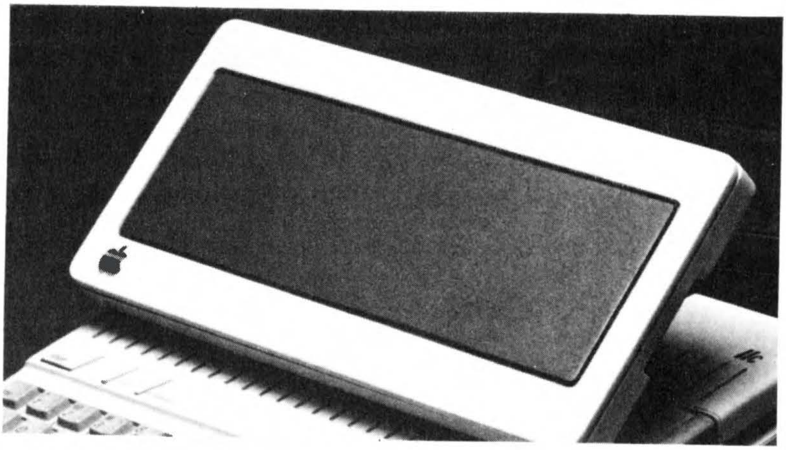
The high-resolution, green and black 9-inch video monitor will display 80 characters per line and 24 lines per screen. It connects to the Apple //c video display interface on the rear surface of the machine. *Apple Computer, Inc* .

Apple //c Flat-Panel Display A lightweight, portable, flat-panel video display for the Apple //c. The display snaps onto the computer behind the keyboard at an optimal viewing angle. It displays 80 characters per line and 24 lines per screen. This display fits conveniently into the Apple //c carrying case and makes the computer truly portable. *Apple Computer, Inc* .

HX-12 The PGS HX-12 is a high-resolution RGB color monitor designed with an NEC .31 mm dot pitch CRT to provide up to 690 dots in the horizontal resolution. The PGS HX-12 delivers 16 supercolors, and 80 characters × 25 lines. *Princeton Graphics.*

NEC 1202 Monitor The NEC 1202 is a high-resolution color monitor. It comes complete with a color kit that includes a cable and plug-in modules for intensity control. *Jack Strick & Associates.*

Sanyo Monitors The following are arranged in order of price:



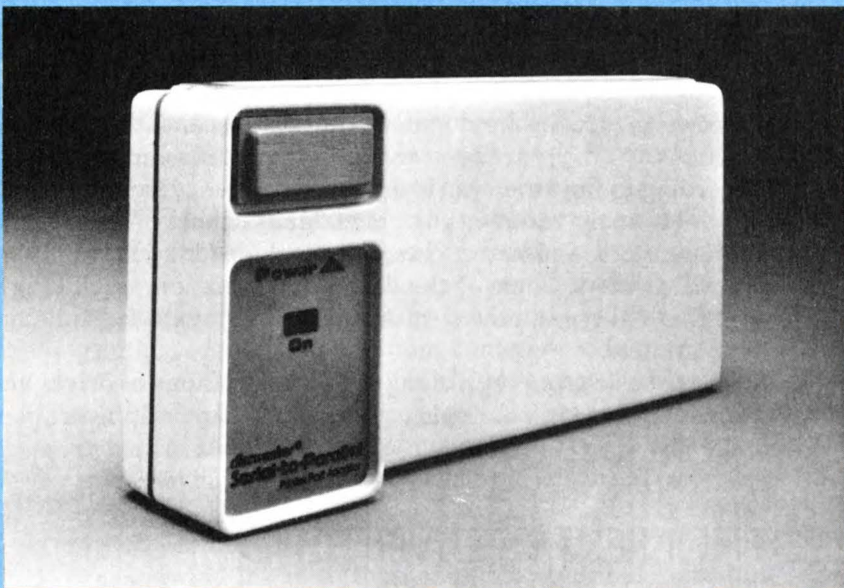
The Apple //c Flat-Panel Display is a revolutionary step in monitor portability. It displays a full 24 lines by 80 characters yet it is very small and lightweight.

9-inch B/W model UM 4509. Compact with a good display, the size is a deterrent for use with an 80-column display.
12-inch B/W model DM5012CX. Good display that is fine for using with 80-columns.
12-inch green model DM5112C4. Same as above, but green phosphor.

Video 7 RGB Adapter This adapter plugs into the TV port of the Apple //c and accepts a standard RGB monitor cable. It contains the circuitry necessary to adapt the Apple //c's video signals so that any standard RGB monitor may be used. *Video 7.*

13

Printers



AN INTRODUCTION TO PRINTERS

A printer is probably one of the first expansion items you will think about for your Apple //c. The following brief discussion should provide the necessary background for you to make a wise choice on which printer to purchase.

In the past, computer printers were generally categorized according to their printing quality. For example, typewriter-like printers, where fully-formed characters strike against an inked ribbon, gave the best quality. These were used for the most important documents, and were known as "letter quality." By contrast, so-called "dot-matrix" printers, where a battery of tiny hammers strike in sequence against an inked ribbon, gave a lesser quality. This category was called "correspondence quality."

However, recent developments have transcended these crude distinctions. Today, printers may be more reasonably categorized according to how they work. For example, the above categories have spawned many variations, such as "daisy-wheel," "line," and "thermal" printers. And newer categories include "ink-jet," "plotter," and "laser" printers. Some of the dot-matrix variations, including ink-jet and thermal types, now surpass the quality of earlier, full-character type printers.

These basic groupings, along with the variations of drives and electronics, provide a complex mix of quality, sophistication, and reliability.

ELECTRONIC TYPEWRITERS

Printers impacting fully-formed characters function similarly to electric typewriters. In fact some electric typewriters, through special intermediate devices, can operate from any personal computer. Typewriters are generally not designed for extended time use with a computer. They are also more prone to jamming with the rapid transfer of data from a computer.

DAISY-WHEEL PRINTERS

One of the more common techniques of producing fully-formed characters uses a set of type elements arranged in a petal configuration. Referred to as "daisy-wheel" printers these devices rotate the mechanism until the correct character lines up for impact. However,

the mechanism slows the printer down. Speeds vary between printers but they usually range from 15 to 40 characters per second.

DOT-MATRIX PRINTERS

A second category of printers forms characters with a pattern of dots. With an impact dot-matrix printer a column of hammers (each smaller than the shaft of a safety pin) are knocked against the paper through an inked ribbon. Only the hammers necessary to form that part of the character actually strike. The column moves slightly and again strikes selectively to form a second part of the character. This process is repeated until the complete character forms. In this way, dot-matrix printers produce characters progressively rather than instantly. The character sharpness depends on the number of dots. Therefore, better printers use more dots. Typical dot-matrix printers use 9 by 9 dots. However, some dot-matrix printers have a higher dot density that produce characters similar to letter-quality printers.

Some dot-matrix printers use thermal technology. Instead of moving hammers, these printers use a row of tiny electrodes and the printing occurs on aluminum-coated or other heat sensitive paper. When an electrode becomes charged, the aluminum burns away to reveal a black surface. Again, the character forms out of a pattern, a matrix of dots. Thermal printers are usually quieter and less expensive than impact printers. However, they need a more expensive type of paper.

Yet another kind of dot-matrix printer uses a row of tiny nozzles for spraying ink onto the paper. These printers are quiet, fast, and can use a variety of colors. They are commonly called ink-jet printers.

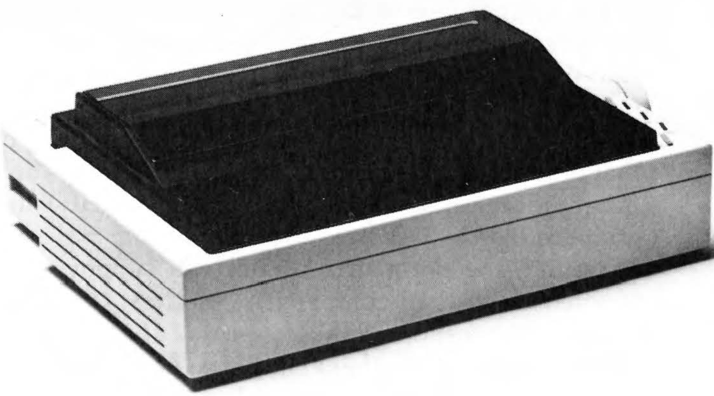
Although the dot-matrix process may seem more complicated than that of letter-quality printers, dot-matrix printers are generally much faster, more versatile, and less expensive. If you intend to print graphics displays, charts, or pictures from programs used by your computer, the dot-matrix printer can support this type of output inexpensively when compared to plotters. Graphics dump software is available to support this feature of the printer. Some programs, such as Apple II Business Graphics, have printer drivers that work with dot-matrix printers. Typical speeds for dot-matrix printers run from 100 to 200 characters per second. Whereas full-character printers are limited to the characters on the print head, dot-matrix printers can often produce several types and sizes of print and graphics with the same head.

APPLE'S SCRIBE PRINTER

Apple's new Scribe printer is a totally new technology. It is essentially a dot-matrix printer, but it uses a thermal method to transfer ribbon material onto ordinary paper. By using a four-color ribbon it can do graphics in four colors. See the brief review at the end of this chapter.

LINE PRINTERS

This category of printers is similar to letter-quality printers in that they produce fully-formed characters. A complete set of characters is contained on either rotating drums or bands that pass above the paper. When a character aligns before the paper, hammers at the rear strike the paper against the character in front. Since each column of the printer has an independent hammer, several characters print simultaneously. This creates an extremely rapid form of printing. Although the characters are fully formed, the high speed often results in a lower quality of print.



The Apple Scribe Printer is a revolutionary new design that transfers ribbon material onto ordinary paper in a dot-matrix format. A four-color ribbon makes color graphics possible.

PLOTTERS

Some printing applications need to produce lines and curves as well as characters. These graphics applications might use plotters. Plotters use a mechanical arm that moves a pen along the paper. The process is relatively slow, and often requires special software, but it can produce detailed and precise drawings. Some plotters may alter the technique by holding the pen still and moving the paper under the pen. These have fewer parts and cost less.

Dot-matrix printers can also produce graphic images. These printers form the lines and curves with dots. Indeed, some plotters may use that technique, but without an extremely high density of dots they cannot emulate the drawing quality of more traditional plotters.

COMPARATIVE FEATURES

The only common thing about printers is that they print. Selecting a printer suitable for your needs requires some knowledge of the features and advantages that go with these capabilities. In the reviews that follow, we will list the features of each individual printer using the criteria described below.

Paper Different printers can accommodate different sizes and types of paper. Thermal printers require a specially treated paper to function properly. Printers with a tractor feed system require a continuous length of paper with a holed, detachable strip along the edges. Detaching the strip may leave an unsightly rough edge. On the other hand, this type of paper can produce lengthy reports without inserting individual sheets of paper. Like an ordinary typewriter, some printers use a friction feed. These may accommodate either separate sheets or a continuous roll. However, continuous rolls usually tend to creep to one edge. Better printers may allow for both types of paper feed systems or present one as an option.

Character Specifications Generally a standard printer can print ten characters per inch. Characters per inch is often referred to as the "pitch." Letter-quality printers may have several replaceable print heads available allowing the user to change the size of the characters printed. Dot-matrix printers can alter a character's size and style through software control. Nearly all printers recognize the full set of ASCII codes. These codes represent the letters of the alphabet, numbers, punctuation marks, and some symbols.

Many printers can print more than one set of characters, such as italic characters, block graphic characters, or special characters used in scientific notation. Some may also include proportional character sets. These position the characters so the space between each character is the same. Some printers can produce hi-res graphics, comparable to plotters while others can underline characters. Some dot-matrix printers have additional modes of operation that improve the print quality. In the emphasized mode, characters are printed twice, the second printing slightly advanced from the first. This fills in some of the spaces in between. The overstrike mode also double prints a line at a time. The paper advances slightly and the line repeats nearly on top of itself. This also tends to fill in the spaces between dots, improving the print quality. Daisy-wheel printers allow the user options in terms of character size and style by providing a variety of interchangeable wheels with character sets of different letter size and style.

Mechanical Specifications Often one of the more important parameters in selecting a printer is speed. Generally, dot-matrix printers are much faster than full-character printers. When dot-matrix printers use some of their features, such as the emphasized mode or double strike mode, the speed drops drastically, usually by one-half. Different character sizes alter the print speed. The speed given in the following reviews reflects the speed of the printer in its standard mode of operation. How fast a printer can advance to the next line is termed paper "slew rate," and the value given reflects the lines per second with a standard 1/6-inch line. Ink ribbons may be a specific cartridge type, available only through a limited number of outlets, or a standard typewriter ribbon. Paper can generally be advanced by hand. Some printers may have a line feed button to advance a single line or a form feed button to advance to the top of the next sheet of paper. A printer may have a switch or button to self-test the printer.

Buffer One feature worthy of special mention is the printer buffer capability. A computer can transmit information to the printer much faster than most printers can print it. Without a buffer, the computer would have to wait for each character to be printed before it could send the next one. A buffer consists of Random Access Memory chips that can store information from the computer. The computer sends information at its own speed and the buffer saves each character sequentially. This buffer storage frees the computer to continue its own operation. The printer continues to print, emptying the buffer. A printer may have its own internal buffer measured by the number of bytes it can store. Many specifically-

designed devices provide printers with buffer capabilities. These printer buffers connect between the printer and computer and must be compatible with both devices.

CONTROL OF PRINTERS

All print data reaches the printer as voltages represented by eight-digit binary numbers. The printer circuitry can recognize those numbers and print the corresponding character. For example, for the letter "A" the computer sends the printer the binary number 00100001 (decimal 65). When the printer receives that number, it takes the necessary action required to print the character "A". All the upper/lower case letters and digits from 0 to 9, have their individual numbers. However,the printer must do more than just print. It must also be able to recognize a carriage return, move the print head back to the beginning, and advance one line. To this end, each of these actions also has an associated number. Similarly, a particular number triggers the printer to skip a space or skip a line.

Some printers have so many features that they require two or more numbers to identify the command. These groups of numbers are sometimes called escape sequences or control sequences. Keep in mind that, while most printers use the same numbers to represent letters and numbers, they do not necessarily use the same numbers to represent escape sequences. To know which codes perform which function, consult the printer manual for a listing or table of which escape or control characters trigger which responses by the printer. Some programs or operating systems rely on programs called "drivers" to control printers. Consult your software documentation to determine whether or not such drivers are required. Programs to control the Apple Imagewriter and Scribe printers are included on the *Aplle //c Explorer's Disk* disk available in conjunction with this book.

BASIC CONTROL OF PRINTERS

Opening Printers In light of the fact that printers respond to appropriate voltages (represented by numbers), it is necessary to have an organized method to deliver those numbers. Additional information concerning the standard used and the port or slot of connection must be provided to the computer. In some cases, as with RS-232 interfaces, additional information as to how fast data will be sent and what form it will take must be included. A program is nec-

essary that will allow the user to implement commands through the keyboard of the computer. Ordinarily you will be using a commercial program that establishes the necessary connection to the printer for you. If you need to do this for yourself in Applesoft BASIC, the command is PR#1. This reflects the fact that the printer is attached to "slot 1" or "port 1."

WORD PROCESSORS

Even with a properly interfaced printer, exerting control through BASIC can become rather complicated. Just getting a character from the keyboard to the printer requires some programming. Fortunately, an abundance of commercially available programs, called word processors, simplify the process. Word processors generally quiz you about your peripherals. For example, they will ask you to state your printer type, and whether it has interfacing. They may even screen a list of some commonly-used printers and ask if yours is on it. Once you reply to the question, the program initiates all of the other programming routines necessary to use your printer. As you enter characters from the keyboard, your monitor displays them. You can then edit or correct your work by altering those screened characters.

When the results satisfy you, a single keyboard command will get them printed. Most commercially available programs such as spreadsheets and databases that use printers work in a similar fashion. These programs also give you a one-time questioning about your peripherals. Sometimes, special print features, like underlining and emphasizing, may require some special handling. Different programs will handle them differently, and some more conveniently than others. The difficulty in utilizing the advanced features of your printer lies with your word processor rather than your printer.

THE INTERFACE

All computers need to tell the printer when a character is ready to be sent, and when it has been sent. The printer in turn must tell the computer that it has received a letter, or that it awaits another one. This signaling back and forth is called "handshaking." The sequence of handshaking signals, along with the format for transmitting data, is called a "protocol."

You could form protocols in an infinite number of ways. Each computer could have its own method requiring a specific printer and

cable. Fortunately, the industry uses three common standards. these standards enable us to connect nearly any printer to any computer. They not only specify how to transfer information, but also what type of cables and plugs to use for connections. Nearly every computer has intermediate devices called interfaces. These connect the computer to the printer to enable the printer to use a particular standard. The Apple //c implements these interfaces through internal circuits connected to the ports in the computer's back panel.

All computers transfer and store information as electrical or magnetic charges. Characters are actually represented by specific patterns of charge. For convenience numbers are equated to each pattern. Each computer could represent characters with its own set of numbers. But again, most printer and computer manufacturers use an industry standard. The most common standard is the ASCII (American Standard Code for Information Interchange).

In selecting a printer you must make certain that the printer uses the same standard that is implemented with your Apple //c's interface—RS-232 serial. Of course, Centronics Parallel printers may also be used with the Discwasher or similar serial-to-parallel adaptor.

STANDARD RS-232 AND PARALLEL CENTRONICS

See Chapter 11, "How to Select Expansion Hardware for Your Apple //c," for a discussion on standard RS-232 and Parallel Centronics.

Below is a glossary of terms used in reviews and descriptions of printers.

ASCII This is a commonly-used list of binary codes used to represent alphanumeric characters. Most printers comply to ASCII code.

buffer Computers can transmit data to a printer much faster than it can be printed. A buffer is a memory device that stores the data until it can be printed. Usually comprised of RAM chips, it may be a feature located within the printer itself or it may be a feature of an interface to the printer. If all the data transmitted from the computer will fit onto the buffer the computer is free to continue with other functions.

condensed mode This mode of operation is used by some correspondence-quality printers to reduce the width of a character to a small size such as 17 characters per inch.

correspondence printers Printed characters composed of dots rather than fully-formed characters are considered to be of lesser quality and not really suited for important letters. Therefore, correspondence may be used to describe the quality of print or the type of printer itself.

double strike This refers to a mode of operation used by some correspondence-quality printers where each character is printed twice in the same location. After a line of characters is printed the paper advances slightly and the same line is printed a second time. This tends to make the impression darker, and avoids the dot-like quality of a dot- matrix printer.

dot-matrix printers A type of printer where characters are printed as a series of dots.

emphasized mode This refers to a mode of operation used by some correspondence-quality printers where each character is printed twice in the same location. After a column of dots is printed, rather than advance for the next column the print head advances slightly and repeats the first column. This tends to make the impression darker, and avoids the dot-like quality of a dot matrix printer.

expanded mode This mode of operation is used by some correspondence-quality printers to double the width size of a character.

font In dot-matrix printers the characters are actually formed from individual dot impressions referred to as fonts. Daisy-wheel printers allow fonts to be varied by changing the type of daisy wheel installed in the printer.

form feed This refers to the ability of a printer to automatically advance the paper to the top of the next page.

friction feed There are a number of methods for inserting and advancing paper in a printer. In a friction feed system, paper is inserted behind a roller like those used in standard typewriters and advanced using the pressure of the roller against the paper.

handshaking When data is transmitted from a computer to a peripheral, such as a printer, the peripheral must send a signal back to the computer to acknowledge that it has received the data. The computer may then send a signal to the printer to advise it that another character is about to be sent. Those signals sent back and forth to prepare for the transmission of data are referred to as handshaking signals.

high-resolution mode With dot-matrix printers and some letter-quality printers it is possible to produce graphic representations or

pictures. The degree of resolution is a function of the number of dots across the width and height.

interface This is a device that connects a peripheral such as a printer to the computer. The interface must be plug compatible with the printer and the computer, and it must be able to recognize the transmission protocol between the two devices. The Apple //c has a serial RS-232 interface built into the case with a five-pin DIN connector on the back. A Centronics Parallel interface is available through the Discwasher or similar serial-to-parallel adapter.

letter-quality Printed characters that are created as fully- formed characters instead of dots have a higher quality of print suited for important letters. Letter quality may be used to describe the quality of print as well as the type of printer itself.

line feed This refers to the ability of a printer to automatically advance the paper one line.

normal This is a standard print size usually accepted as ten characters per inch.

on line This indicates that the printer is connected to the computer and data is free to be transmitted.

off line This indicates that data is not free to be transmitted to the printer. Either it is not physically connected or a switch has been set to prevent data from being transmitted.

paper out light This feature indicates that the paper has run out. Usually the printer will immediately disconnect from the computer and wait until paper is inserted before continuing to print.

protocol When a computer attempts to transmit data to a peripheral, such as a printer, it must send the data in a format that the printer will recognize. After each portion of data is transmitted, the printer must be able to communicate back to the computer that it has received the data and is ready to receive the next portion. Both the printer and the computer must agree with the signals that are sent back and forth. The format with which data is transmitted and the sequence of signals sent back and forth is referred to as protocol. Some standards, but by no means the only types of protocol, are RS-232, Centronics Parallel, and IEEE. The built-in printer port on the Apple //c uses a serial RS-232 protocol.

self-test A feature found on some printers that tests the printing capability of a printer. When a printer fails to print, a self-test will help determine if the problem is in the printer or in the computer.

slew rate This is the rate at which paper can be advanced in a printer. Usually it is measured in lines per second.

subscript This feature indicates that a printer can print condensed characters in width and height at the bottom of a line.

superscript This feature indicates that a printer can print condensed characters in width and height at the top of a line.

tractor feed There are a number of methods for inserting and advancing paper. A tractor feed system contains rotating sprockets that insert into holes along the edges of special paper.

The following table provides a basis for comparing dot-matrix printers available for the Apple //c. Each of those printers has a list price under $1000.

Name	Price Range	Speed CPS	Multiple Fonts	Dot Addressable
Anadex DP-9000A	H	150	Y	Y
Axiom GP-250	L	50	Y	Y
Axiom IMP-4	M	100	N	Y
C. Itoh 8510 BCD	M	120	Y	Y
C. Itoh 1550 BCD	H	120	Y	Y
Data Impact 81A	M	100	Y	Y
Data Impact 92	M	100	Y	Y
Dataproducts 480	M	70	N	Y
Epson RX-80	L	100	Y	Y
Epson FX-80	M	160	Y	Y
Heath 125	H	150	N	Y
Mannesman Tally MT 160	H	160	N	Y
Mannesman Sprint MT 80	M	80	N	Y
NEC PC8023A	M	120	Y	Y

Price ranges: L=under $550, M=$551 to $750, H=$750 to $1000. More expensive printers are available, including designer and gold-plated models.

The following printers and related devices will work with the Apple //c in their serial versions, with an appropriate cable or adapter to fit the Apple //c's five-pin DIN connector for the serial printer port.

Apple Color Plotter This small, easy-to-use plotter can create colorful charts, graphs, tables, and drawings for business, scientific, or personal use. The Color Plotter can handle up to four colors at once and up to eight colors total by changing pens when necessary. You can create graphic documents on paper or overhead transparencies of various sizes because of the wide plotting bed. Parallel versions will require a Discwasher or similar serial-to-parallel convertor.

The Apple Color Plotter comes with a muilt-pen Capper kit, power cord, serial interface cable, eight pens in assorted colors, paper and instructions. Other supplies are also available. The Plotter connects to serial I/O port 1 on the rear surface of the Apple //c. It works well with Apple Business Graphics and other graphics software. *Apple Computer, Inc.*

Apple Daisy Wheel Printer This letter-quality impact printer will print a variety of professional documents created on your Apple //c computer. It has many useful features: it produces highly-readable letters at a speed of 40 characters per second; it offers a variety of type font print wheels including Courier 10-pitch, Prestige 12-pitch, Gothic 15-pitch, and Executive Boldface; foreign language (such as French, German, and others) print wheels are available; it will print large documents with up to 198 characters per line; and it allows for proportional spacing, variable horizontal and vertical formatting, forms length selection, and automatic form feed.

A bidirectional, tractor forms feeder is also available to print up to six copies and handle forms from two to 15 inches wide.

The Apple Daisy Wheel Printer comes with a power cord, the standard print wheel and ribbon, and assembly instructions. It is attached to serial I/O port 1 on the rear surface of the Apple //c. *Apple Computer, Inc.*

Apple Imagewriter Printer For home or office use, this dot-matrix printer will print documents and graphics at up to 120 characters per second. Text can be printed in upper and lower case. Proportional spacing and six different form lengths can be used. The Imagewriter can print seven different character styles in eight different sizes. High-resolution graphics are printed at 160 by 144 dots per square inch. Friction-feed or sprocket paper between 4½ and 10 inches can be used.

The Imagewriter will work with both of Apple's new machines—the Apple //c and the Macintosh. One switch inside the printer must be set properly to make the printer run if it had been set up for a Macintosh. These switches are inside the printer's cover on the bottom of the printer on the right hand side, covered by a clearly labeled fold-back plastic cover. The switches are in two blocks numbered SW-1 and SW-2. Switch number 5 on switch block SW-1 must be in the closed position to allow for data to be passed from the Apple //c to the printer. The closed position is clearly marked, and you can flip the switch with a pen or letter opener. Be sure you change the correct switch, or you will introduce still another problem. The open setting of this switch sets up the printer for use with Macintosh. The *Apple //c Explorer's Disk,*

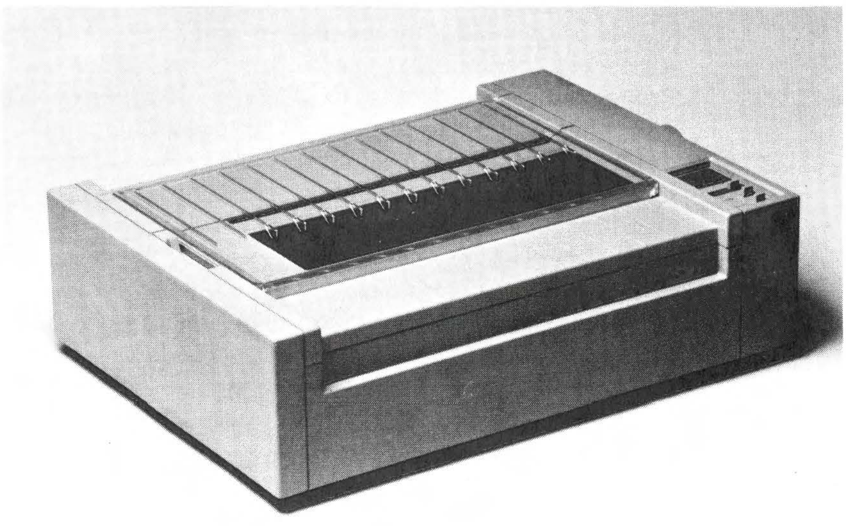
available in conjunction with this book, provides a program to control the advanced features of the Imagewriter printer from easy-to-use menus.

Hopefully the dealer who sells you the printer will set the switch correctly and try out the printer for you, but if the printer does not work this is at least one possibility you can check.

The Apple Imagewriter Printer comes with a power cord, ribbon, and assembly instructions. It is attached to serial I/O port 1 on the rear surface of the Apple //c. *Apple Computer, Inc.*

Apple Scribe Printer A thermal-transfer printer that works with regular paper because it heats the special ribbon onto the paper rather than heating the paper itself.

The Scribe Printer can print in two modes: graphics and character. It is uni-directional, has a 4K buffer for speedy printing, prints upper and lower case characters, and prints up to 80 characters per second. High-resolution graphics can be printed at up to 160 by 144 dots per square inch. In graphics mode, the special four-color ribbon allows color printing in yellow, magenta (deep red), cyan (dark blue), and black. It can handle friction-feed or sprocket paper. The *Apple //c Explorer's Disk*, available in conjuction with this book, provides a program to control the advanced features of the Scribe Printer from easy-to-use menus.



Apple's Imagewriter Printer is the standard printer for Macintosh. It does an excellent job on text and graphics with the Apple //c.

The Scribe attaches to serial I/O port 1 on the rear surface of the Apple //c. *Apple Computer, Inc.*

Buff-et This is a high-speed buffer for parallel printers. It will hold an average of 30 pages of data. Available in three sizes—16, 32, or 64K—it also comes with the ability to test itself and send out messages to the printer. Another handy feature is the ability to make multiple copies of a document without having to send it to the buffer a second time. *Renaissance Technology Corporation.*

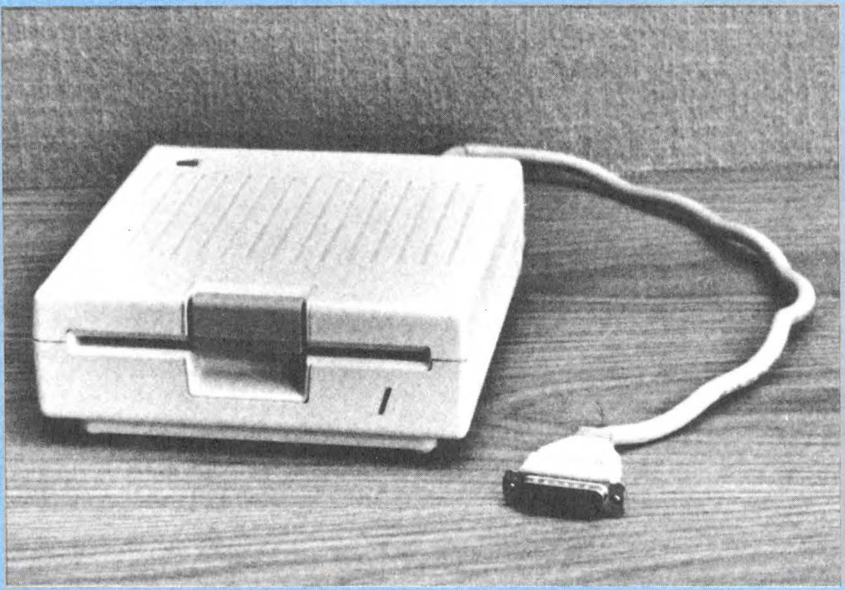
Microprism Microprism is a low-cost printer with 75 CPS operation, pin and friction feed, and data printing at 110 CPS. It contains both RS-232 and parallel interfaces, and features data plot graphics. *Integral Data Systems, Inc.*

Serial-to-Parallel Printer Port Adapter This adapter attaches to the serial printer port and accepts a Centronics Parallel printer cable. See the picture on the chapter opening page.

Strobe Model 100 (drum type plotter) A hi-res plotter that accepts many different pens, has an interactive digitizing mode, is easy to use, and permits the X-Y coordinate data corresponding to pen position to be entered directly into the computer. Other features include variable character sizes, vector plotting, horizontal and vertical character strings, and axis generation. The operator control includes PEN LIFT control, ENTER/START command switch, and two-speed positioning in the four directions of motion. *Strobe, Inc.*

14

Disk Drives



AN INTRODUCTION TO DISK DRIVES

The Apple //c comes with a built-in single-sided disk drive, located on the right side near the back. This disk drive is your primary method for storing data and programs. Disk drives are random access devices. If you wish to copy the tenth of ten programs into your computer you can copy it directly without having to go through the first nine. This ability to randomly access information from the disk makes it possible to sort and search information quickly and efficiently. Consequently, many applications involving searching and sorting may only be possible because of the storage offered by the disk.

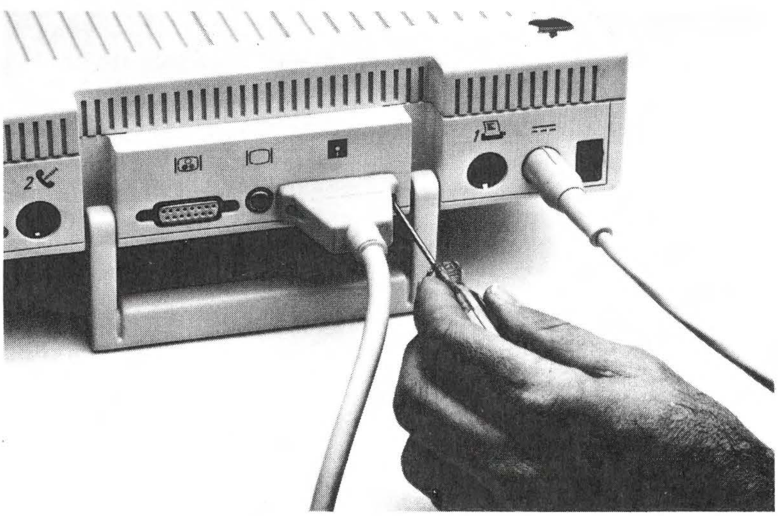
You may find the built-in disk drive adequate for all your needs, or you may want to add an additional drive. This is as simple as plugging in a toaster, provided you have chosen a disk drive compatible with the Apple //c. The connector for the second disk drive is already on the back of the Apple //c. The only thing you need to check is that the drive is Apple II compatible, and that the connector on the cable will fit the connector on the back of your Apple //c. Larger disk storage is available on hard disks such as the Quark QC 10. To really gain an understanding of how computer systems work it is important to gain an insight on how disk drives work.

ProDOS—THE DISK OPERATING SYSTEM

The program on a disk is permanent. You could turn the disk drive off, store the disk on a shelf for a month, then return and copy information from the stored disk into the computer's memory. The copy you make into memory of the Apple //c is temporary. As soon as you turn the computer off the program is lost. If you wish to run the program again you will need to make another copy from disk into your computer. A disk drive is called a permanent storage device.

As a permanent storage device a disk drive must accomplish a variety of tasks. It must be able to store a program on a disk. It must be able to find that program and copy it to the computer upon your request. In addition to programs that control the computer information generated while using a program may be stored on a disk drive as well. A page of text entered from the keyboard, or a list of numbers, or the results of computer calculations may be stored on disk.

This type of information is stored in particular formats called files. The disk drive must be able to recognize and handle each type of file.



An Apple //c compatible, external disk drive connects in seconds and can save time and tribulation with disk operations.

A disk drive is managed by an integrated circuit called the disk controller. The Apple //c's disk controller is an advanced single-chip design called an "Integrated Woz Machine" or IWM. The disk also requires a set of programs called the Disk Operating System that work together to provide the control that will direct the drive to perform each of its tasks. The disk operating system supplied with the Apple //c is ProDOS, an advanced operating system that gives you a choice of ways to perform disk operations.

ProDOS accepts instructions for disk operations as machine language interface calls. Since constructing these calls is rather complex, two simpler interfaces are provided to make ProDOS easy for you to use. The first is a program called BASIC.SYSTEM, which makes ProDOS look very much like the older Apple DOS 3.3. This is a great service to those who are already familiar with DOS 3.3, and provides a very simple way to use ProDOS from BASIC. At this level commands such as CATALOG, CREATE, RENAME, DELETE, LOAD, SAVE, and others may be used to specify disk functions. BASIC.SYSTEM converts these easy-to-use commands into the MLI calls required by ProDOS. Some of these commands are described in Chapter 8—Applesoft BASIC, and complete details are available in *BASIC Programming With ProDOS.*

The second simple interface to ProDOS is provided by a program, which is stored on the ProDOS *System Utilities* disk. This program

provides the set of menus that make ProDOS functions available to even the beginning computer user. No knowledge of programming is required to use ProDOS through these menus, since the menu program gathers all the information it needs through its menus and formats the calls to ProDOS to do the required disk operations. This level of use of ProDOS is covered in Chapter 5—System Utilities.

HOW THE DRIVE AND DISK WORK

The 5¼-inch floppy disk looks like a flat, black square paper-like object with a hole in the middle. Actually that paper-like material is a vinyl jacket for a circular piece of mylar plastic inside. The mylar is coated with microscopic crystals of a metallic oxide. The mylar disk is very smooth, flexible, and free to rotate within the vinyl shell. The tiny crystals of metallic oxide on its surface can be magnetized by placing a magnet or a device that creates magnetic fields near the disk. Just as a metallic object can be magnetized by holding it near a strong magnet, portions of the disk can be magnetized in the same way.

If you have ever played with two magnets you are aware that the two ends of the magnets behave differently. One end attracts a particular end of another magnet and repels the opposite end. One end is the "north" pole of the magnet and one end is the "south" pole. Portions of a disk can likewise be magnetized with the north or south pole oriented in a particular direction. In addition, an area of the disk may be magnetized either weakly or strongly or anywhere in between. It is this ability to magnetically charge portions of the disk with varying north/south pole orientation and varying intensity that enables us to store information on a disk. Every number or letter that is stored on a disk is stored in this way.

A computer stores information using voltages and magnetic orientation (north vs south). A single voltage source is either on or off and is called a bit. Eight bits are grouped together to form a byte and are used to represent alphanumeric characters and other data (binary numbers, machine language programs, and others). A byte of data can be transferred from the internal memory of a computer to the disk drive through wires connecting the two devices. In the disk drive the data is converted into a series of magnetized portions on disk of varying intensity.

Information can be read off the disk in a reverse process. The read/write head is passed along the surface of the disk. As it encounters the portions of the disk that were magnetized in the writing process, voltages are induced in the head and applied to bits in

the drive's memory. The drive circuitry decodes the magnetic information to reconstruct the data that was originally written. The data is transferred back to the computer where it is available in the same eight-bit byte form as it was originally written from.

The vinyl jacket has a small notch along one edge. Placing a piece of tape over this notch write-protects the disk. With the tape in place information can still be read from the disk but nothing can be written onto the disk. Some disk drives have switches that allow write-protection tapes to be overridden.

TRACKS, SECTORS, AND DIRECTORIES

Information is written onto a disk in an organized fashion. ProDOS organizes the information so that it can find it again when needed. The Apple //c uses soft-sectored disks that are unformatted when first purchased. In order to establish the required framework for storing and retrieving data from the disk, it must be put through a special one-time process called formatting. This function is available from the *System Utilities* disk. Formatting the disk creates the tracks and sectors that will be used throughout the life of the disk (until it is reformatted, which destroys any data on the disk). Formatting places information on the disk that allows the disk controller and ProDOS to determine where the read/write head is located. It also establishes an area on the disk, called the volume directory, where ProDOS will keep track of files on the disk as they are created, modified, or deleted. This volume directory reflects the name and location of files on the disk, some of which may in turn be subdirectories pointing to still other files. You can see a formatted display of some of the contents of the volume directory of a disk by choosing the "Identify and Catalog a Disk" option on the *System Utilities* disk. If you want to snoop into the details of ProDOS disks on a sector-by-sector, byte-by-byte basis, use ProSNOOP from the *Apple //c Explorer's Disk* available in conjuction with this book.

Data is written onto a disk in concentric circles called tracks. When a disk is inserted into a drive, a gripper clamps down on the small portion of exposed mylar in the hole in the middle and begins spinning the disk within its vinyl shell. Another part of the disk drive called a read/write head rests against the disk through the elliptical window in the shell. As the disk spins the read/write head is free to extend to and away from the center of the disk and in this way can access any part of the disk surface. As the disk spins, the head can generate magnetic fields to write data as positively or negatively charged portions of the disk with varying magnetic intensity and/or

north/south orientation. The head may begin near the outside edge of the disk and write data as the disk spins beneath it. When the disk makes a complete rotation of 360 degrees, the head will again be positioned at the starting point. The head then extends slightly towards the center of the disk and begins writing a new track of data.

Information on a track is further organized into sectors and blocks. Each track will be divided into a number of sectors with each sector containing a certain number of bytes of data. The number of tracks and sectors a drive uses varies with the type and model. The built-in disk drive of the Apple //c uses 35 tracks each containing 16 sectors of 256 bytes (with Apple DOS 3.3 and ProDOS). ProDOS further groups the 256 byte sectors into 512 byte blocks and stores files in whole blocks. The closer a drive squeezes tracks and the bits on sectors, the greater the amount of information that can be saved on a disk. But squeezing tracks and bits together requires more sophisticated components on the drive and a better quality of disk. Drives have been designated as either single-density drives or double-density drives depending on how tight they squeeze information together on a disk. A double-density drive is theoretically capable of storing twice as much information in the same space as a single-density drive. Some drives squeeze the tracks even closer together—hence a quad-density drive. Double- and quad-density drives require a superior disk and the quality of a disk is also designated as either single-, double-, or quad-density.

When a program is saved on disk, the Disk Operating System (either DOS 3.3 or ProDOS) records the name of the program and the track and sector where the program is placed in a special location on the disk called the directory. When you save a program to disk you are required to assign a name to the program or file. The Disk Operating System will automatically find some free space on the disk, save the program onto the disk, and enter into the directory the name and starting track and sector. If you wish to find that program again and copy it into your Apple //c's internal RAM, all you need to do is give DOS or ProDOS the name of the file. The Disk Operating System will automatically find the name in the directory, position the head over the starting location, and read the information off disk.

Even after the head is positioned over the correct track it must find the point where each sector begins. It must be able to distinguish between the beginning and the end of a track. There are two approaches to solving this problem used by most drives. The earliest drives used a method called hard sectoring where a series of holes separating each sector were put into the disk. A light placed on one side of the disk and a sensing device on the other allows the drive to recognize the beginning of each sector. A second method, used by

the Apple //c's disk drives, is termed soft sectoring. Instead of a hole in the disk, information written on the disk when it is formatted is used to determine where data is stored on the track.

When a disk comes from the manufacturer it is essentially blank. Before the first program can be saved onto it, all of the special characters to indicate the beginning of each track must be put onto the disk by a process known as formatting the disk or initializing it. Once a disk is formatted and the sectors marked, the disk is essentially marked for life. If you attempt to reformat the disk it will introduce a new beginning for each track and sector and any programs on the disk will be written over and lost.

Since hard sectoring requires a disk with prepunched holes a hard sectored drive will require a disk designated as being hard sectored. The basic mechanics of the process described is the same for all disk drives. The number of tracks created on a disk may vary from drive to drive. Some drives have two heads that access both sides of the disk simultaneously. Disks for these drives must have windows on both sides of the disk and are called double-sided disks.

The bottom line of all these variations is that generally you must obtain disks designed for the Apple //c drive you are using—single-sided, dual-density, soft-sectored. And if you decide to add a second, external disk drive, you must obtain a disk drive designed for the Apple II family of computers with a cable that will connect to the Apple //c's DB-19 external disk drive port.

The following terms are used in reviews and descriptions of disk drives.

block Two sectors, or 512 bytes of information, stored on a disk. This is the basic unit of information used by both ProDOS and the UCSD p-System operating systems.

directory An area of the disk set aside to keep track of the names and locations of files located on a disk.

disk Disk refers to the media used for permanently storing information from a disk drive. In the strictest use of the word a disk is a rigid plate-like object whose surface can be magnetized to store information. More general use of the word includes the floppy disk, which is made of a flexible material.

disk drive A device that contains the mechanism for writing and reading information to or from a disk.

disk A disk appears as a square, paper-like object with a hole in the center. The paper houses a circular piece of plastic that is free to rotate within the paper shell. The surface of the plastic is coated

with metallic oxide. Portions of the surface can be magnetized to store information permanently.

DOS Disk Operating System. Disk drives perform a number of functions. At the very least they can write, read, and erase information from the disk. The Disk Operating System is a set of programs residing either in the disk drive, a controlling board, or in the computer's memory that provide the control for performing those functions. Apple DOS 3.3 and ProDOS are the two major Disk Operating Systems for the Apple //c. ProDOS is more powerful, flexible, and easier to use than DOS 3.3. ProDOS is supplied with the Apple //c and will certainly be the main Disk Operating System for most users. The only reason you would need to be concerned with DOS 3.3 is if you have existing DOS 3.3 files or programs or want to use material developed for DOS 3.3.

double-density Information is stored on disk by magnetizing microscopic portions of the disk. The quality of the disk and the read/write heads in the drive limits how close these microscopic portions can be before they start interfering with each other. Double-density disks are of a higher quality and, with the correct disk drive, can store twice as much information as a single-density disk. The Apple //c uses double-density (single-sided, soft-sectored) disks.

double-sided The quality of the surface of a disk is a vital factor in storing information accurately. A double-sided disk has had the quality of both sides of the disk verified before release. Some disk drives can use both sides of the disk to store data. The Apple //c uses single-sided disks. Both sides of the disk can be used by cutting out an extra write-protect notch on the disk, turning it over, and writing data on the bottom side. Such a disk is called a "flippy" disk. This does not make it a double-sided disk–it is one disk being used to hold two single-sided disks worth of data. Only one side at a time is available for use. To access the opposite side of the "flippy," it must be manually removed from the drive, flipped over, and re-inserted. For true double-sided disks and drives, two read/write heads are required and both sides are available for data storage at the same time, without removing the disk and flipping it over. "Flipping" disks is not officially endorsed and disk manufacturers will not guarantee disks so used unless they were originally certified for double-sided use.

flippy See Double-sided.

floppy A word used synonymously with disk. It refers to the flexible nature of the plastic disk and housing.

format An initialization process used on most disks that establishes tracks and sectors on the disk and usually also creates the volume directory.

hard-sectored Information is stored on disk by magnetizing microscopic portions of the disk. These microscopic portions lie on concentric rings called tracks, which are subdivided into sections called sectors. It is vital that the disk drive know exactly where a track and sector begins and where it ends. Hard sectoring refers to a process where holes through the disk are used to indicate where a sector and track begins. The Apple //c uses soft sectoring.

load A term used to describe the entry of information into a computer from an external device such as a disk drive.

random access A random access device is capable of selecting a program, or files, or even a particular part of a file, and reading it immediately. Contrast this with a sequential device that must first read all the programs or files that come before the selected program.

read When information is copied from a permanent storage device into the memory of a computer the process is called reading.

save A term used to describe the storing of information from a computer onto a permanent media such as a disk.

sector Information is stored on disk by magnetizing microscopic portions of the disk. These microscopic portions lie in concentric rings called tracks. Each track is subdivided into sections called sectors. The Apple //c's built-in drive stores 16 sectors of 256 bytes on each of the 35 tracks on a disk.

sequential device Some devices by their nature must read and write information in consecutive locations. A cassette, for example, stores bits of information one after another. To read a file near the end of the tape every file prior to that must be read first. (The Apple //c does not support a tape cassette.)

single-sided The quality of the surface of a disk is a vital factor in storing information accurately. A single-sided disk has had the quality of one side of the disk verified before release. It is intended to be used with disk drives that access only one side of the disk. The Apple //c uses single-sided disks.

soft-sectored Information is stored on disk by magnetizing microscopic portions of the disk. These microscopic portions lie in concentric rings called tracks and are subdivided into sections called sectors. It is vital that the disk drive know exactly where a track and sector begins and where it ends. Soft-sectoring refers to

a process of writing special characters onto each sector and track to keep track of the beginning and end. The Apple //c uses soft-sectored disks.

track Information is stored on disk by magnetizing microscopic portions of the disk. These microscopic portions lie along imaginary concentric rings on the disk called tracks.

write When information is copied from a computer's main memory into a permanent storage device, such as a cassette or disk, the process is called writing.

write-protect Most disks contain a small notch in one of the edges. If this notch is covered with a piece of tape it becomes write-protected. That is, you can continue to read information from that disk but you cannot write information onto it. An effort to write onto the disk will result in a write-protect error.

The following disk drives and related products are available for the Apple // family of computers. Only the Apple Disk //c is certain to work correctly, have correct cables, and comply with all government regulations concerning TV and other radio interference. Check for other drives that are certified to work with the //c. We have listed here some drives from independent companies that we think are likely to be adapted for use with the Apple //c, but you must be sure to check for versions specifically adapted to the //c's and to find out how special software needs are to be met. The simplest solution for extra disk storage is to go with the Apple Disk //c, which is designed for and matches the Apple //c in style. Be sure to notice the Quark QC10 ten-megabyte hard disk designed to work with the Apple //c.

A-2 An exact replica of Disk II, 140K. *Micro-Sci.*

A-40 A 40-track version of the A-2, 171K. *Micro-Sci.*

A-70 A double-sided drive with 286K of storage. It is Disk II compatible. *Micro-Sci.*

AMDISK-I This unique drive uses a three-inch cartridge disk (not compatible with other Apple drives). It provides 286K of storage and is compatible with the Apple Disk II. *Amdek.*

AMT Drive A high-quality yet low-cost 163K drive. This half-height drive is considered extra reliable. The mechanism is by Panasonic. *Advanced Micro Technology.*

Apple Disk //c This half-height, single-disk drive, capable of storing 140K of data, was designed to be used as an add-on external drive for the Apple //c. This drive has the same streamlined

styling as the Apple //c and uses the same closing mechanism as the internal drive.

There are several advantages to adding a second drive to your Apple //c. A number of sophisticated business software packages require a second drive to run on the Apple //c. Also, you have faster, more convenient access to data without having to change disks. *Apple Computer, Inc.*

C-111 A Disk II compatible 163K drive. *Concorde.*

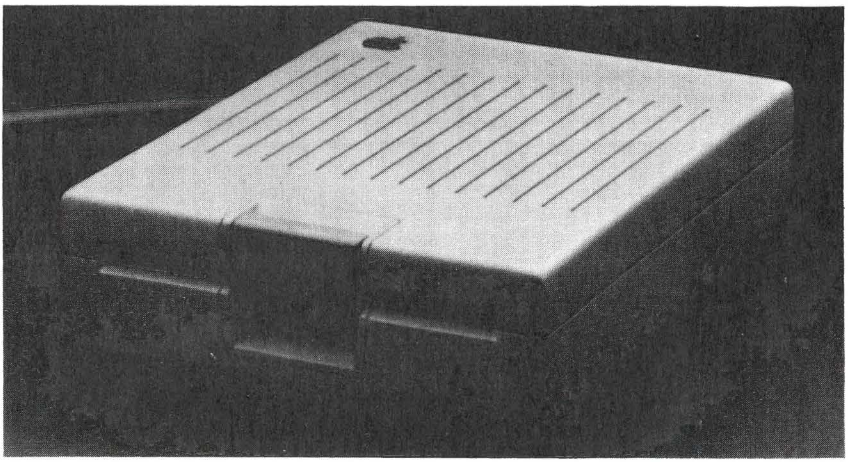
Disk II This is the standard 140K Apple drive. It will not be used on the Apple //c because of cable and TV interference requirements. It is replaced by the slimline Apple Disk //c for use with the Apple //c. *Apple Computer, Inc.*

DiStar An inexpensive half-height drive with 160K storage. *Burke and Associates.*

Elite-One This drive provides a 40-track, 163K alternative to the Apple Disk II. It is hardware compatible but requires special software that is included. *Rana Systems.*

Elite-Two This drive is similar to the Elite-One and Elite-Three. It gives you 320K of storage on a double-sided 40-track floppy. *Rana Systems.*

Elite-Three This drive provides 652K of storage using quad-density dual-sided technology. It is hardware compatible with the Apple Disk II but requires special software that is included. The



The Disk //c is specifically designed to match the Apple //c electronically and in appearance. It uses a half-weight drive, so it is much smaller than the older Apple Disk II drives.

Elite-Three can read standard Apple disks, but the Apple Disk II cannot read disks written on the Elite-Three. *Rana Systems.*

Franklin 10 A copy of the Disk II, 140K. *Franklin Computer Corporation.*

Mate-I A totally compatible 5¼-inch floppy disk drive for the Apple II. Memory capacity and track formatting are identical to the Apple's, so no modification of software is necessary. *American Mitac Corporation.*

Micro Sci Drives There are three Apple-compatible drives available: the A2 (143K), A40 (164K, 40 tracks), A70 (286K). *Micro-Sci.*

Mini Taur A half-height drive compatible with the Disk II, 140K. *RGB Designs.*

QC10 A ten-megabyte hard disk for the Apple //c. The QC10 plugs into the external disk port. You can still plug another floppy disk drive into the QC10. *Quark Peripherals.*

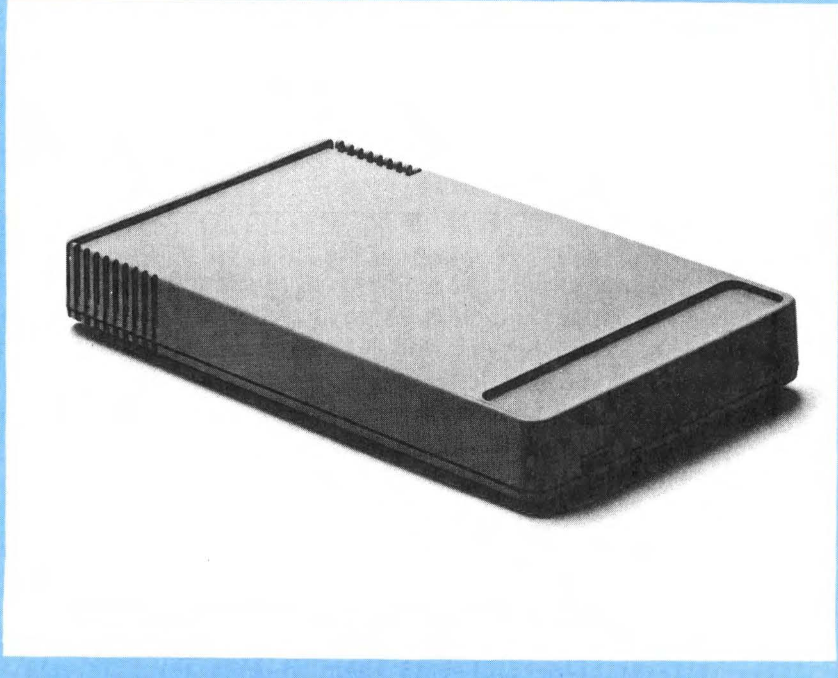
Solo An inexpensive, totally Disk II compatible drive. *Vista Computer Company.*

Taur II An Apple Disk II lookalike, 140K. *RGB Designs.*

Taur II+ A 171K single-sided drive. *RGB Designs.*

15

Modems



AN INTRODUCTION TO MODEMS AND TELECOMMUNICATIONS

Perhaps one of the most exciting aspects of the Apple //c is its ability to talk to other computers. By a process called "telecomputing" you can access information from large systems through your home phone. With the proper equipment you can dial a number on your phone, connect it to your computer, and begin sending or receiving information from another computer thousands of miles away. Some of the information and services available today include up-to-the-minute news and prices on stocks, articles from encyclopedias, reviews of books and movies, home banking, and bill-paying services.

There are local services called bulletin boards where you can read about meetings and products for sale. There are dating services and ride boards. Some services provide electronic mail. You enter through your keyboard messages that are saved on another computer for someone else to read at their convenience. For professions that put you in and out of an office during the day, electronic mail offers a convenient and efficient way to receive and return messages.

The U.S. Postal Service and MCI both offer a computer postal service. You type in a letter through your computer and they will print it and send it anywhere in the country. MCI combines this with delivery services that promise four-hour delivery times to most major cities.

Many banks are now offering home banking and bill-paying services. You can see an up-to-the-minute statement on your accounts, transfer funds between accounts, and pay bills from the convenience of your living room.

Some companies are looking into the possibility of having employees work at home and send their work in through their home computers.

You can take a college math class, play chess with someone in another city, or purchase a piano by using the keyboard of your computer.

With nearly all of these services there is an initial fee in establishing an account with the service. You then pay an additional charge for the time you use the information service. The fees may range from a few cents to a few dollars a minute depending on when and what service you access.

The Source (1616 Anderson Rd, Mclean VA 22102) is one of the largest information services available. Billed as an "information utility" they provide a daily news summary, abstracts from financial

publications, financial reports on over 3,000 companies, bond and commodity prices, and movie and restaurant reviews. Through The Source you can purchase items from an electronic catalog, play games, and send electronic mail to other members. Establishing an account costs $100, plus $7.75 to $20.75 an hour for connect time. Some hardware and software manufacturers include either free or discounted memberships to The Source (and/or CompuServe, Dow Jones, or other services) the purchase of communications-related items.

The Dow Jones News/Retrieval (P.O. Box 300, Princeton, NJ 08540) provides stock prices on companies on all the major exchanges. Closing prices can be obtained for any company for previous days, months, and years. Financial disclosures can be obtained for most companies. Up-to-the-minute news affecting particular companies or categories of companies can be accessed. Articles from the Wall Street Journal and Barrons can be accessed as well as information from a 20-volume encyclopedia. There are various connection fees. Some modems include a free subscription to the service. Connect time ranges from $.15 to $1.20 per minute.

Dialog's Knowledge Index (3460 Hillview Ave. Palo Alto, CA 94304) provides summaries and full text of over seven million articles, reports, and books on technical topics. Dialog also provides a listing of over 25,000 publications available through the U.S. Government Printing Office. Establishing an account costs $35 plus $24 per hour of connection time.

CompuServe Information Service (5000 Arlington Centre Blvd., Columbus, OH 43220) provides a number of investor-oriented services including stock quotes, historical data on more than 40,000 stocks, and business data from Standard & Poors and Value Line. CompuServe also has a number of services directed to assist computer users. One service through CompuServe will allow you to ask technical questions concerning your Apple computer. Establishing an account may cost $40 or be included with the purchase of some modems. Connect time ranges from $5 per hour to $22.50 per hour.

The list of services and opportunites through telecomputing is growing and the expense for such services is constantly dropping. For an excellent coverage of available databases and related services, see *OMNI Online Database Directory* by Mike Edelhart and Owen Davies (Omni Publications International, Ltd., 1983).

There are four requirements in telecomputing–your Apple //c, a modem, a phone, and a program. You need a modem, a device that connects your computer to the phone, which makes it possible to transmit and receive information across regular phone lines. You need access to an ordinary telephone that will physically connect

you with distant computers. Finally, you need some specialized software that will allow you to enter information from your keyboard and view information across your screen.

Each of the hardware devices must be compatible with each other. The modem must be designed to work with your Apple //c. The program must likewise work with your computer and modem. Again, there are standards for data communications and, there are just enough variations within the standard to make it important to make your selections carefully.

THE MODEM

Most modems for personal computer systems comply to RS-232 standards, compatible with the built-in modem port (serial port 2) of the Apple //c. Data is sent serially one bit at a time. The RS-232 standard defines voltage levels and a system of handshaking, but leaves how data is actually sent across a phone line to the design of the modem.

Most of the initial work on modems began at Bell Telephone Laboratories. The most common type of modem for home use emulates the Bell 103 type. This type uses the RS-232 standard for data protocol but further prescribes how data is converted to frequencies and sent across the phone line. Two separate frequencies are used to send data: one at 1070 Hz and one at 1270 Hz. If a transmitting computer shifts from one frequency to the other, the computer at the receiving end can detect the change and interpret the shifts as bits of information. With Bell 103 modems a different pair of frequencies, 2025 and 2225 Hz, are used to return information. The same method of frequency shifting is used to transmit one bit of information at a time. With information transmitted on one frequency and received on another it is possible to send and receive information simultaneously. This ability to transmit and receive simultaneously is referred to as full duplex capability. Most commercial information services for personal computers provide for interaction with Bell 103 type modems. There are a number of commercially available modems of this type.

Bell 103 modems are limited to 300 baud rates. That means it takes about two minutes to receive one typed page of information. There are other types of modems that will transfer data at a faster rate. Some modems, including those that emulate the Bell 212, transmit and receive at 1200 baud—nearly four times faster than the Bell 103 type. These modems are generally much more expensive but may pay for themselves in the reduced time you spend

accessing information. Another type of 1200 baud modem is the Racal-Vadic, which is much less common than the two Bell-compatible types.

There are modems available that will automatically dial a number that you have prerecorded. These autodial modems will simplify the process of connecting with an information service you may use frequently. There are also autoanswer modems that can be used to receive electronic mail from other computer users. With your computer left on and the appropriate software left running, an autoanswer modem will automatically answer the phone and make the connection that allows data to be transmitted to your computer.

THE PHONE CONNECTION

The modem attaches to the Apple //c through the built-in modem port at the back of the machine. The modem must still be connected to the telephone. There are two main types of connections with phones. The first uses an acoustic coupler, a device that allows you to fit the handset of the phone into a cradle on the modem. You need a phone with a handset and a modem with the acoustic coupler. An acoustic coupler is well suited for someone who travels and connects a modem to pay phones often. The second type allows you to disconnect the jack of a modular telephone and insert it directly into the modem. This requires the use of a modular phone, which is becoming the standard in most homes in this country.

THE SOFTWARE CONNECTION

A computer with a properly connected modem is still incomplete. A special type of program is needed to begin telecomputing. This type of software is sometimes referred to as communications software, terminal software, or terminal emulators. Regardless of the name the function is the same. It enables you to enter information from your keyboard and send it through the modem. The same software must take information that has been sent from a distant computer and display it on your screen. The software must be written for the Apple II family of computers, or for the Apple //c.

Although the RS-232 standard prescribes a method of handshaking, whereby two devices know when data has been sent or

received, it leaves a number of questions unanswered. A number of parameters concerning how data is transferred must be decided upon before data can be exchanged. Computers at both ends of an RS-232 connection must agree on how fast data is to be transferred. The rate, measured in bits per second, is called the baud rate. Software at both ends must be designed to establish a constant rate of transmission. If the baud rates are not the same at both ends one device will receive data faster than it can process it. This will result in lost data. Computers must also agree on duplex. Most information services require a full duplex capability where data can be sent and received simultaneously. But some may require half duplex capability where data is sent and received over the same lines but not at the same time.

Computers exchanging data through communications must agree on parity. Some computers add up the first seven bits of a byte of data. Using an even parity check they set the eighth bit to whatever is necessary to make the total sum even. A computer at the other end of a transmission checks to see that the sum of the bits in a byte is even. If it is not there must have been an error in transmission, and it requests that the byte be sent again. Computers must agree on an even parity scheme, an odd parity scheme, or to ignore the parity scheme entirely.

Computers must agree on word length before transmission can occur. Normally eight bits are used to represent a character, but in some cases seven bits can be used and the eighth bit ignored. Each byte of data in RS-232 transmissions must be framed with start and stop bits. An extra bit or two is needed to indicate the end of one byte of data and the beginning of another. Both computers must agree on whether one or two bits are to be used. With most information services the parameters described above have been established. You must make the same settings on your computer.

Good communications software available commercially will make the settings automatically for you. They will prompt you for baud rate, parity, word length, duplex, and stop bits. You simply answer the questions and the program does the technical work required to set up a connection with the remote service or computer. Some communications software will do a lot more. Some programs will store the information that you access through your telephone in a process termed downloading. You may then save the information stored onto diskette or have it sent to a printer. Uploading allows you to send information already in your computer through your modem and phone to a receiving computer.

It is often necessary to send a series of account numbers and passwords to an information service before they will allow you to access information. This initial preparation is called logging on and

varies with each service. Some communications software allows you to permanently store all the account numbers and passwords and automatically transmit them whenever you wish to log onto a service. Such a prestored sequence of commands, log-on sequence, and passwords is often called a "macro." Some modems designed for the Apple //c come with enough software to allow you to communicate with remote computers, but for many modems there exists a number of programs that may be purchased separately to enhance the modem's capability.

With most modems the procedure for establishing contact with a remote database is the same. You dial the telephone number for the information service. You will hear the phone ringing at the other end. Suddenly you will hear a steady high-pitched tone indicating that the distant computer is ready to establish contact. You must then connect your modem to your phone using one of the two methods described above. With the appropriate software in place, you should now be able to log onto the host computer and begin exchanging information. With an autodial modem, this procedure is automated. All you do is specify or select from a menu the number you want. The software and the modem do the rest.

SUMMARY

To enable your Apple //c to communicate with distant computers, the first step is to obtain a compatible modem. The next step is to obtain communications software specifically for the Apple //c. Most modems include a program for the Apple //c. These programs are loaded and executed from diskette just like any other program. There are also commercially available programs that can be purchased separately, which provide additional features that may not be found on the software that comes with the modem. Once the equipment is in place and the program running, it is simply a matter of calling the information service. There are over 2,000 remote databases that provide some sort of service. Each database may require that your computer comform to certain RS-232 parameters. These parameters include the baud rate, the start and stop bits, the parity, the duplex, and the word length. Most communications software will prompt you for the value parameters and automatically set them. Once the phone is answered at the other end a steady high-pitched tone will be heard. On hearing the high-pitched tone the phone is connected to the modem and communication begins. Each service will have its own set of commands for accessing information.

The following terms are frequently used in reviews and descriptions of modems and related products.

acoustically coupled A modem attached to the handset of a telephone is acoustically coupled.

ASCII American Standard Code for Information Interchange. A code used by many computers to represent characters of the alphabet, numbers, and punctuation marks.

asynchronous Serial data may be transmitted using a number of protocols. There are two general ways to handle the transmission of serial data, synchronous and asynchronous. With asynchronous transmission both the transmitting and receiving devices agree on the rate that data will be transmitted. They also agree on special characters or bits to indicate the beginning and end of the data bytes—the start and stop bits.

baud rate The data transmission rate of a device. It is usually measured in bits per second.

Bell 103 A type of modem that uses RS-232 protocol, but it also specifies such parameters as duplex and maximum baud rate.

BPS bits per second.

buffer A temporary intermediate storage area for data.

control character A byte of data that has a special meaning. It may be used to indicate the beginning or the end of a transmission or to initiate a response at the receiving end.

database A collection of information and data. A remote database refers to an information source that must be accessed by some means of telecomunications.

duplex There are two forms of duplex. Full duplex indicates that a computer can transmit and receive information simultaneously. Half duplex indicates that a computer can only perform one of those functions at a time.

echoplex With some transmissions information is automatically returned and displayed for the sender. This process, called echoplexing, provides a method of error checking and verifies that data is transmitted to and from a secondary device corrrectly.

modem A term derived from modulator demodulator. This device converts digital information from a computer into analog information that can be transmitted across telephone lines. It also converts analog information back into digital information.

parity Some computers check for transmission errors by counting the number of "on" bits in the first seven bits of a word. Even

parity is created by setting the eighth bit to whatever is necessary to make the sum total of "on" bits equal an even number. Odd parity is generated by setting the eighth bit to whatever is necessary to make the total odd. A receiving device can compare the parity bit with the total sum to check for an error that may have occurred during transmission.

remote computer This refers to a computer that is located so that it must be accessed with the use of telecommunications equipment.

RS-232 This refers to a specific way of transmitting data between devices. The voltages and handshaking protocol are defined and data is sent serially one bit at a time. This standard is used with most commercial modems and some printers.

serial When information is transferred between two devices, one bit at a time, the transmission is said to be serial.

start bits When information is transferred serially in an asynchronous method, additional bits indicating the beginning of each byte of data must be supplied. These additional bits are called start bits. Stop bits indicate the end of each byte.

stop bits See start bits.

synchronous Serial data may be transmitted using a number of protocols. There are two general ways of handling the transmission of serial data, synchronous and asynchronous. With synchronous transmission a special sync mark or character is initially transmitted to tell the receiving device that data is about to be sent. It may also indicate how much data will be sent. The receiving device receives a clock signal that accompanies the bits of data to synchronize the reception of data.

telecomputing The exchange of information between two or more computers through the telephone lines.

word length The number of bits in a unit of data—usually seven or eight.

The following modems and related products are available for the Apple //c. Some will require a special cable or adapter to fit the serial port connector of the Apple //c. All of the 300 baud modems will work with the //c. For 1200 baud modems, check with the retailer or manufacturer to be sure you are getting an Apple //c compatible version.

APPLE / / c Compatible Modems.

Company	Product	Price	Type	Bell Prototype	Operations	Voice	Baud Rate	Duplex	Dialing	Special Features
Anderson Jacobson	AJ4048	H	DC	None	O/A	Y	4800	F	None	Error correction; 4800 duplex
	1212-ST	L	DC	103, 113, 212	AA	Y	300, 1200	F,H	None	None
	1212-AD1	M	DC	103, 113, 212	O/A	Y	300, 1200	F,H	P,T	Stores 16 numbers up to 37 digits each; integral speaker
	1212-AD2	M	DC	103, 113, 212	O/A	Y	300, 1200	F,H	P,T	Auto log on; two level security
Apple Computer	Apple Modem 300	M	DC	103	O/A	N	300	F,H	P,T	LED function, computer 300 AA external speaker
	Apple Modem 1200	M	DC	103, 212	O/A	N	300, 1200	F,H	P,T	LED function, 1200 external speaker
AT & T	103JR	H	DC	103	O/A	N	300	F	P	None
Bacus	AC 312	L	AC	103	A	N	300	F	P,T	BASIC unit
Bizcomp	1012 Intelli-Modem	M	DC	103, 212	O,A	N	300, 1200	F,H	P,T	Repeat dial on busy signal; programmable auto log on; full self-test; auto speed detection
	1022 Intelli-Modem	M	DC	103	O,A	N	300	F,H	P,T	None
Bytcom	212AD	L	DC	103, 113, 212A	O/A	N	300, 1200	F	P,T	Two-year warranty
Campbell Scientific	DC103A	L	DC	103	A	N	300	F	P,T	Battery operated

APPLE //c Compatible Modems (continued).

Company	Product	Price	Type	Bell Prototype	Operations	Voice	Baud Rate	Duplex	Dialing	Special Features
Cermetek	Infomate	M	DC	212	O/A	Y	300, 1200	F	P,T	Electronic call progress tone detection; stores 52 32-character phone numbers
Codex	5103	L	DC	103, 113	O/A	N	300	F	P,T	Local analog loop-back; remote digital loop-back
	5202	L	DC	202	AA	N	1200	F,H	P,T	Local analog; local self-test; remote self-test
	5212 Autocall Unit	M	DC	103, 113, 212	A,O, MO	N	300, 1200	F	P,T	Auto redial; single keystroke dialing; end-to-end self dialing; local analog; remote digital
Coherent Communications	Linemate 96	L	DC	None	O/A	Y	300	F,H	None	Simultaneous voice and data
	SPM-94	L	DC	None	O/A	Y	300	F,H	None	Simultaneous voice and data
Comdata	212E2-32	L	DC	212A	O/A, MO,A	N	1200	F	P,T	Can be used on two-wire leased line
	305E2-12	L	DC	103	O	N	300	F,H	P,T	Voice data switch; two-wire leased line
	370E2-42 Phonem	L	DC	103J	O/A	N	300	F,H	P,T	One-number dialer; auto activate exclusion key
Computer Development Corp	ET	L	DC	103	AA	N	300, 1200	F,H	P,T	Voice/data switch; optional encryption chip
	ETC	H	DC	212A	AA	N	300, 1200	F,H	P,T	Printer buffer; optional encryption chip

APPLE //c Compatible Modems (continued).

Company	Product	Price	Type	Bell Prototype	Operations	Voice	Baud Rate	Duplex	Dialing	Special Features
Concord	212	M	DC	212	O/A	N	1200	F	P,T	Auto adaptive equalizer
Develcon Electronics	6212 Smartmodem	L	AC	212A	AA	N	1200	F	P,T	Speed dialing; battery backup; user-friendly commands
	8212	M	DC	212A	O/A	N	300, 1200	F	P,T	Autodial
Digital Equipment	100 Modem Family	M-H	DC	None	AA	N	1200 9600	F,H	P,T	Neither terminal or system dependent
Gandalf	Sam 212A	M	DC	103, 212	O/A	N	300, 1200	F	P,T	Supports 7-, 8-, and 9-bit codes; auto log on; stores 52 32-number characters
Hayes	Smart-Modem 300	L	DC	103	O/A, AA	N	300	F,H	P,T	LED function; external speaker
	Smart-Modem 1200	M	DC	103, 212A	O/A	N	300, 1200	F,H	P,T	LED function; external speaker
Incomm	A-1200	L	DC	212A	AA	N	1200	F,H	P,T	None
	Starcomm	L	DC	103, 212A	AA	N	300, 1200	F,H	P,T	RS-232 cable included; Hayes-compatible
Inmac	8071	M	DC	212A	AA	N	300, 1200	F	T	Test lights on front panel; Hayes-compatible
Integrated Design Engineering	1200 Baud Modem 8070	L	DC	212, 212A	O/A	N	300, 1200	F	T	Includes RJ-11 cable
	Smart Talk 300 Baud 8063	L	DC	103	AA	N	300	F	T	Can be used with standard phone jack

APPLE //c Compatible Modems (continued).

Company	Product	Price	Type	Bell Prototype	Operations	Voice	Baud Rate	Duplex	Dialing	Special Features
MFJ	MFJ-1232	L	AC	103	O/A	Y	300	F,H	P	Can use AC or battery; works on TTL as well as RS-232
Microcom	RX-1000	M	DC	212A	AA	N	300, 1200	F,H	P,T	Error correction; auto redial
Micom	MB80514	M	DC	212A	AA	N	300, 1200	F	P,T	No battery backup required
Multi-Tech	MT212AH (Multi-modem)	M	DC	212, 212A	O/A	Y	300, 1200	F,H	P,T	Auto-dial with commands from keyboard; detects dial tones as well as distant ringing and busy signals
NEC	N212DR	M	DC	103, 212	AA	N	300, 1200	F	P,T	Built-in number directory; security password
	N103JR	L	DC	103	AA, MA	N	300	F	P	Rack mountable
Novation	Smart-Cat 103/212	M	DC	103, 212	O/A	N	300, 1200	F	P,T	Analog loop-back; programmable
Prentice Pop-Com	X-100	L	DC	103, 113, 212A	AA	Y	300, 1200	F	P,T	No hardware switches; Hayes-compatible
Processing Innovations	Speech- Aided Modem	L	DC	202	AA	N	1200	H	P,T	Digital voice synthesizer; voice verification
Prometheus	Promodem 1200	L	DC	103, 212A	O/A	N	300, 1200	F,H	P,T	Help command from keyboard; clock on board; internal diagnostics
Racal-Vadic	VI 1222	M	DC	None	O/A	N	1200	F	P,T	None
Radio Shack	DC 1200	M	AC	212	AA	N	1200	F,H	T	Multi-line controller

APPLE / / c Compatible Modems (continued).

Company	Product	Price	Type	Bell Prototype	Operations	Voice	Baud Rate	Duplex	Dialing	Special Features
Rixon	R103J	L	DC	103, 112, 212	O/A, AA	N	300	F,H	P,T	LED indicators; push button test
	R212A Intelligent	L	DC	103, 113, 212A	AA	N	1200	F	P,T	Memory battery protected
Tri-Data Oz	Guardian 533	M	DC	103, 212	AA	N	110, 300, 1200	F	P,T	Security password
Universal	103 LP O/A	L	DC	103, 113	O/A	N	300	F	P,T	Line powered
U.S. Robotics	Password Modem				O/A	1200	P,T			
Ven-Tel	1200 Plus	L	DC	103, 212A	A	N	300, 1200	F,H	P,T	LED lights; internal speaker; detects incoming calls
Wang	WA3451	H	DC	212A	O/A	N	300, 1200	F	T	None
Western Datacom	212	M	DC	103, 212A	O/A	N	300, 1200	F,H	P,T	None
	Worldcom 200	L	DC	202	O/A	N	300, 1200	H	P,T	European protocol supported

Price: L = under $500; M = $500 to $1000; H = over $1000

Voice: Y = Yes; N = No

Type: DC = Direct connect; AC = Acoustic coupler

Duplex: F = Full; H = Half

Dialing: P = Pulse; T = Tone

Operations: A = Answer only
AA = Auto answer
MA = Manual answer
MO = Manual originate
O/A = Originate/answer
O = Originate only

The following are brief reviews of some of the modems available for the Apple //c.

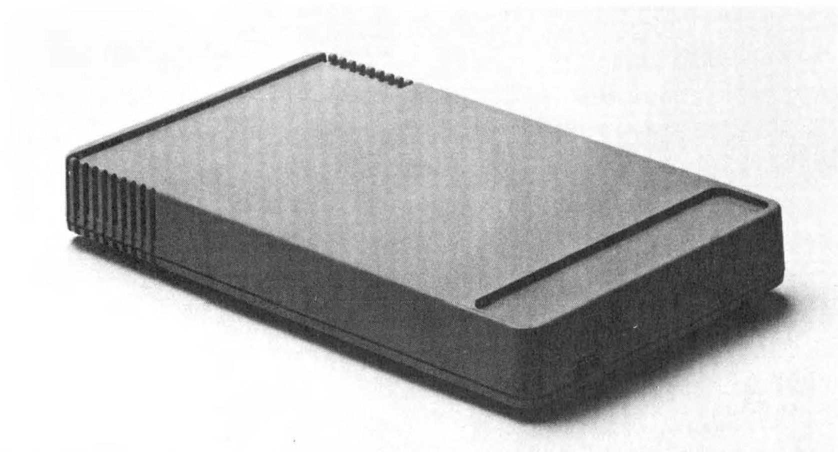
Apple Modem 1200 A microprocessor-controlled modem capable of transmitting data at up to 1,200 baud (120 characters per second). It is controlled from the Apple //c keyboard with simple keystroke commands. This modem features autodialing and autoanswering.

The Apple Modem 1200 plugs into serial I/O port 2 on the rear surface of the Apple //c. The modem comes with telephone interface cable, serial data cable, power cord, and manuals. *Apple Computer, Inc.*

Apple Modem 300 A microprocessor-controlled modem capable of transmitting data at 300 baud (30 characters per second). It is controlled from the Apple //c keyboard with simple keystroke commands. This modem features autodialing and autoanswering.

The Apple Modem 300 plugs into serial I/O port 2 on the rear surface of the Apple //c. The modem comes with telephone interface cable, serial data cable, power cord, and manuals. *Apple Computer, Inc.*

Auto-Cat A low-cost, autoanswer, direct-connect modem approved by the FCC. Auto-Cat gives you 24 hours of answering service and data communication. When the business executive is on vacation, Auto-Cat will let the executive "talk" with the com-



The Apple Modem 300 is convenient and attractive. It is Hayes Smartmodem compatible and works with Access II and most other communications programs for the Apple //c.

puter while computer enthusiasts can access their computer by phone from any location. Auto-Cat is compatible with any Bell 100 series modem. The low-profile case fits under your telephone and all controls are easily accessible. The power supply plugs directly into wall sockets. *Novation, Inc.*

Datalink Word of mouth has it that, "Nothing communicates like Datalink," so you many want to look into this excellent piece of equipment. Just push one button and Datalink will dial you in and log you on. Should you need to redial, Datalink will do it for you automatically. It works with almost any modem used with the Apple. *Link Systems.*

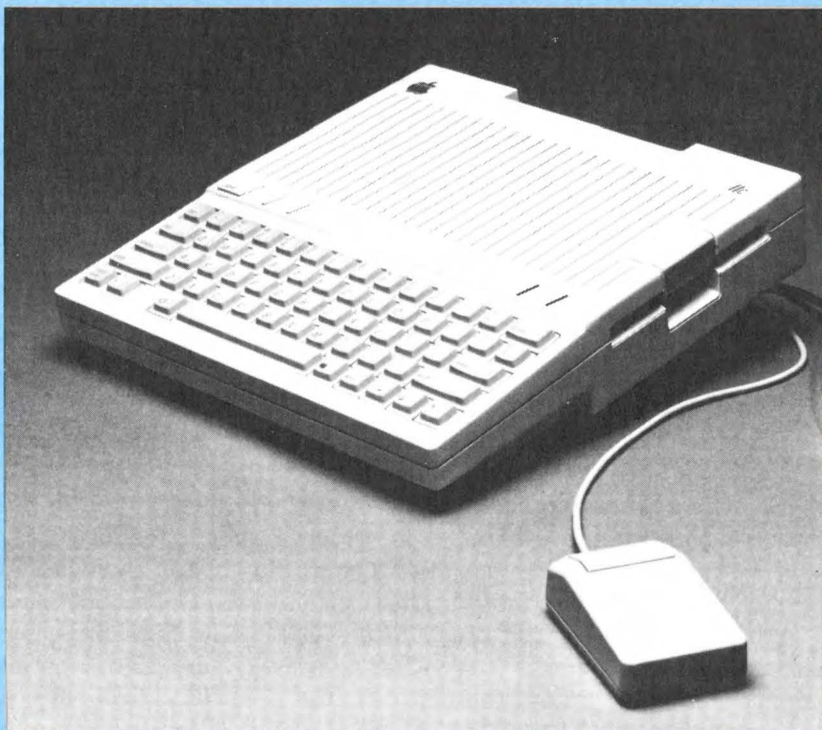
D-Cat A directly-coupled modem that is easy to use, inexpensive, and portable. This is the only direct modem that the FCC has approved for handset jack connection with modular phones. It transmits data over telephone networks; permits one computer to "talk" with another; and data exchange is very quick (30 characters per second). Attractive, easily installed, and equipped with power plugs, this is another winner from Novation. *Novation, Inc.*

Hayes Stack Smartmodem An RS-232C autoanswer, autodial data communications system compatible with virtually all personal computers. Standard features are touch-tone or pulse dialing and direct connection to single or multi-line telephones. The system can be program controlled by any language through ASCII character strings. Operating parameters are changed with ease by a series of unique "Set" commands and eight configuration switches. Smartmodem's operating level is full- or half-duplex up to 1,200 baud. Seven LED indicators on the front panel provide a visual check of system status, and the user can monitor the progress of calls via an audio speaker. The system features automatic baud rate, parity sense, and word size detection. A two-year limited warranty covers the system. *Hayes Microcomputer Products, Inc.*

Signalman Mark I Modem The Signalman Mark I Modem has "reinvented" the modem, packing the "state-of-the-art" technology into the most compact, versatile modem on the market today to offer you maximum performance. *Leading Edge Products, Inc.*

16

Mice (Mouses?)



INTRODUCTION TO MICE (MOUSES?)

A mouse is a small, palm-sized "box on wheels" that is moved across a flat surface that controls the movement of the cursor. Usually, the mouse is placed on the desktop to the right or left of the computer and moved around as desired. Buttons on the top of the mouse are used to select functions depending on the number of times or the order in which they are pressed. Cursor movement and other functions can be performed considerably faster with the mouse than with the traditional cursor movement keys.

The first mouse was developed in the early 1960s by Douglas Englebart, while working at Stanford Research Institute on a project to find interactive computer aids. It was described as a small mouse-like object with buttons "sticking up on top like ears." The wheels mounted underneath measured movement along the x- and y- axes, which was then converted into movement of the cursor onscreen. Englebart and his associates used the mouse successfully at the Institute for a number of years.

The mouse was largely ignored for many years, being considered a gimmick by many people. Only lately have people realized that the mouse can significantly enhance the speed and accuracy of many tasks involving a computer. In several comparison tests, the mouse scored high in word processing when used against the light pen, joystick, conventional keypad, and digitizing tablets. They also were less fatiguing to use than the new "touchscreen," because the user's arm did not have to be held up high.

There are two main types of "mice," mechanical and optical. The mechanical mouse, which is the older of the two technologies, works by counting the revolutions of a shaft attached to the wheels or ball in the base of the mouse. This may be done electrically or with an optical decoder. The mechanical version has numerous, precision moving parts and is susceptible to breakdown due to dust and grit contamination from the desktop. The precision parts, also, tend to make the mechanical mouse more difficult to manufacture, expensive, and very delicate.

The optical mouse has no moving parts and is, therefore, easier and less expensive to make. They track their position optically, by passing over an optical grid and counting the lines. The scale of movement can be changed simply by changing the size of the grid. The resolution of the optical mouse is not as fine as the mechanical mouse, but is sufficient for most applications. One drawback is that the optical grid must be kept on your desk when the mouse is in use, increasing the "footprint" of the computer.

Unlike a digitizing tablet, which uses absolute positioning, the mouse uses relative positioning. In absolute positioning, there is a one to one correspondence between points on the surface of the tablet and points on the screen. Relative positioning, however, is based on the total amount of movement of the positioning device, without reference to where it started or where it stops. For example, if you are trying to move the cursor across the screen and run into the side of the keyboard before you can reach your destination, simply pick the mouse up, reposition it to allow sufficient room, and continue moving the mouse. The cursor will start moving from the point where you left off, in the same direction as the movement of the mouse.

The number of applications that make use of the mouse is growing daily. Word processing is a hot new area. Several new full-featured word processors, featuring the mouse as an integral tool, are now on the market. While the mouse is useless for typing in the first draft of a document, the editing process can be enhanced tremendously with the mouse. The buttons can be programmed to perform specific editing functions and the cursor can fly about the screen locating editing changes much more rapidly than is possible with the Apple //c's cursor control keys. Spreadsheets now make efficient use of the mouse in much the same way as word processors. Graphics packages that allow the user to select, move, enlarge, and paint shapes are becoming very popular. Freestyle drawing is also possible with the mouse. Creating and revising complex graphics can be done in record time. Even programming can be enhanced using the mouse.

The following are brief descriptions of the AppleMouse and MousePaint, which are available for the Apple //c. Other mice, compatible with the Apple //c, may appear soon.

AppleMouse // This is a mechanical mouse that moves over the desktop on a small rubber ball. It requires about a 12-inch square work area. The four-foot cable that attaches the mouse to the rear of the computer allows you to position it on the left or right side of the keyboard. The mouse connects to the nine-pin port on the rear surface of the Apple //c. The large square connector at the end of the cable screws in easily by turning the two knobs on each side.

The AppleMouse has one button on top that is used to issue commands by moving the pointer to the various menu options, and to draw shapes by holding the button down while moving it across the screen.

The AppleMouse comes with a graphics program called MousePaint. This program allows you to create drawings, diagrams, charts, and lettering. It is menu-driven and uses icons to represent the various options. The options include a sketchpad, which provides

an area to draw on, and drawing tools such as a pencil, spray can, brush, letter, straight edge, and eraser. The entire picture or large portions of it can be manipulated using these editing functions: move, cut and paste, copy, delete, invert, flip, and undo. Pictures created with MousePaint can be stored on disk or printed.

MousePaint is a fun way to learn how to use the mouse and it can create useful graphics for business or home use.

The AppleMouse. Small, simple, elegant, yet powerful with software like MousePaint. AppleMouse brings some of the power of Lisa and Macintosh to the Apple //c.

IV

Appendices

Appendix A
Books and Magazines

BOOKS

Academic Apple, The By Richard Mowe. This is an unusual book. It is written primarily to assist adults helping children learn by using the Apple. Expectations of what a child should be able to do at various ages, from pre-school through high school, are addressed in the chapter called Getting Started. After discussion of the use of commercial software, Mowe then goes into programming and the requirements for designing your own software. The final section addresses learning to type and using the computer as a word processor. The appendices contain helpful forms, a listing of helpful books and magazines, sources of software evaluations, software recommendations, and a glossary. The reading level is grade 6. *Reston Publishing Co.*

Apple BASIC for Business for the Apple II By Alan J. Parker and John F. Stewart. This is primarily a college text. There are problems at the end of most chapters, but there are no answers included. Presumably they are available in a Teacher's Guide. The primary emphasis is on business applications, so the use of files is introduced as early as the fourth chapter. Both sequential and random access (the text calls them direct access) files are taught, with all the types of manipulations needed for an office. It would be helpful, in using this book, to have a previous knowledge of BASIC, although some of the commands/statements are given in the third chapter, and a summary of BASIC commands and instructions are given in the first appendix. *Reston Publishing Co.*

Apple Graphics Games By Paul Coletta. The book begins with an explanation of shape tables, surely one of the more abstruse concepts in the Apple world, and continues with sound generation routines and an alphanumeric character set. You then are shown how to develop a menu that will access any of the games found later in the book. The reading level grade is 9.5. The games are fully listed in the text so that you can key them in. An optional diskette containing all the games in the book is written in Applesoft BASIC, consisting of the following:

Match is a timed game for one player who tries to move a shape from the edge of the screen to superimpose it upon its match.

Piano simulates that instrument by composing, playing, and saving tunes.

Pairs is a two-player game similar to Concentration, with a couple of bonus-point extras.

Catch uses the game paddles. You play Perry Como and "catch a falling star and put it in your pocket."

Boxes is connect-the-dots without your having to draw the original pattern. Naturally, it is a two-player game, although playing both parts might be a way to raise your self-esteem as a winner!

Poker is five-card stud, aces high, for two players.

Equate requires you to find the correct set of numbers to complete an equation. Easier said than done, as the numbers are hidden in a grid, and you must locate the adjacent numbers with the game paddle before your opponent does.

Spiral draws spiral designs using BASIC instead of Turtle Graphics in Logo where this type of design is most often found.

Lander is a lunar lander game for one player. You score higher if you get onto a smaller landing pad.

Spider involves moving a fly about a spider's web, eating spider eggs. (Pac-Fly??). *Reston Publishing Co.*

Apple Logo Primer By Gary G. Bitter and Nancy Ralph Watson. The book is spiral-bound so that it lies flat for easier use during the learning process, which is a great help to someone who only has two hands, both of which are needed on the keyboard. The first section is a tutorial, almost complete, for someone who has never used Logo, consisting of 14 chapters moving from Turtle Graphics, through procedures, the REPEAT function, editing, recursive programs, input and variables, arithmetic operations, conditionals, animation, text, and combined text and graphics.

In the quick-start second section are the directions for someone who has used another version of Logo and wishes to become familiar with Apple Logo in the least amount of time. The third section, Applications, is a discussion of the major versions of Logo. Appendices contain answers to the practice activities, a list of suggested activities that should keep any Logo student occupied for a long time, reference information for turtle commands, file commands, screen commands, editing/control characters, common error messages, arithmetic operations, and Apple color codes. The reading level is grade 7. *Reston Publishing Co.*

Apple Pascal, A Hands-On Approach By Arthur Luehrmann and Herbert Peckham. This tutorial book for learning Pascal allows anyone with the Pascal diskettes (four are required for all the files) to learn programming in Pascal. The lessons are developed for use with one disk drive, although two drives may be used later on. Each chapter begins with a set of goals and ends with a summary. There are

problems at the end of the chapters that may be answered by using the computer. *McGraw-Hill Book Co.*

Apple User's Encyclopedia By Gary Phillips, Joyce Conklin, and Donald J. Scellato. This book offers a comprehensive coverage of the Apple II family and of general personal computer terms and concepts. It is an excellent overall reference source. It does not cover the Apple //c specifically, as it was published before the //c was released. Most of the material applies to the //c as a member of the Apple II family. *The Book Company.*

Beneath Apple DOS By Worth and Pieter Lechner. This book is the classic explanation of the esoterica, minutiae, and details about how Apple DOS works. In addition, with a little bit of study, it is reasonably easy to understand. If you want more detail on DOS, VTOC, track organization, DOS program logic, customizing DOS, and other facts, this is the best reference for you. *Quality Software.*

Better BASIC for the Apple By J.N.P. Hume and R.C. Holt. This text, written by two professors at the Department of Computer Science at Toronto University, addresses the subject of BASIC programming explicitly for the Apple. The sequence of chapters is reasonably standard, with summaries and exercises at the end of each. The discussion of structured programming found here is not present in most other texts. The answers to the exercises are *not* included in the text. Presumably, they are available from the publisher. The reading level is grade 8.75. *Reston Publishing Co.*

Database Management for the Apple By Nat Wadsworth. This book supplies an extensive discussion of the theory and operation of database management and its applications. The advantage to using this text is that all the routines are written in Applesoft and thus load and sort in seconds or minutes. This might not be what you are looking for, but if you want to understand how files and other portions of databases operate, this gives you a good start. It also has a chapter on customizing your database and adding "bells and whistles." The final topic is expansion of the memory-resident system (presented in the book) into one that will use random access files, thus expanding the storage capabilities and allowing access to the disk during the run of the program. The book includes line-by-line commentaries, a complete cross-reference of used variables, and a list of cross-referenced line numbers. *Hayden Book Co. Inc.*

Graphics Cookbook for the Apple By Nat Wadsworth. The purpose of this book is to present an explanation of a method to construct shapes and pictures using low-resolution graphics that does not require add-on machine language subroutines. In keeping with the

title, there is a selection of 100 pictures that include background scenes, objects, shapes, and forms. Both illustrations (looking very much like patterns for cross-stitch embroidery, due to the use of a 40 X 40 grid) and data statements containing the x- and y-coordinates are provided. The actual method is a BASIC program that reads the coordinates from the data statement and implements them in the colors indicated by the main program.

Also included is a section on creating a low-resolution character set, which would be especially useful in primary education programs. While some of the pictures are rather simple (a knife, a small flying saucer, and a small pipe), others are considerably more complex (border #3, background #2, a butterfly, and a fireplace) and are well worth the time required to key them in and save them as text files for later use in programs. The reading level is grade 8. *Hayden Book Co. Inc.*

How to Write an Apple Program By Ed Faulk. If you were expecting this book to teach you Applesoft BASIC, try again. On the other hand, if you want a book to discuss program design, sources of program ideas, coding, and debugging for maximum usefulness, then this is your book. All of the sections are illustrated by a CHECKBOOK program that is developed from beginning to end. The book contains amusing cartoons, lighthearted but easy-to-understand writing, and a bibliography of additional resources. If you are in a hurry for a checkbook program, the entire code is listed in one place in an appendix. *Datamost, Inc., Reston Publishing Co.*

Microcomputer Use & Software Design No. 7884 This is a booklet for educators. It is filled with ideas about ways to introduce students to microcomputers. Includes a glossary of terms. *Milton Bradley Educational Software.*

Programmer's Handbook to the Apple II This is a paperback book edited and compiled by Conway B. Christensen, published by Computer Station. It covers the Apple //c as well as its predecessors. The book is a collection of useful material from the *Apple II - DOS Manual, Applesoft II, Apple 6502 - Assembler/Editor, Apple II - Apple Plot, Apple II - Applesoft Tool Kit, AppleWriter, Apple II - Reference Manual, DOS 3.2 Version, Pascal, VisiCalc,* and *Microsoft Z-80 Softcard Vol 1 & 2.* It uses material from each of these manuals and references to explain how the Apple computer works, how it is programmed in various languages, and how to use a number of programs such as Applewriter and VisiCalc.

The function of this particular book is to provide a complete, portable reference for the veteran programmer to carry from place to place or refer to rather than carrying all of the manuals. Its treat-

ment of all material is brief but complete. It first presents a discussion of the System Monitor and how to use assembly language. It then covers DOS and DOS commands. Then it covers the Apple 6502 Assembler/Editor and both versions of BASIC used by Apple computers. The book then covers Pascal, the Macro-Sced screen editor, the DOS Tool Kit, Applewriter II Word Processor, VisiCalc (16 sector), Microsoft CP/M, and BASIC 80 (the Microsoft version of CP/M BASIC). The final portions of *Programmer's Handbook* are devoted to Apple computer hardware descriptions, Silentype, and Paper Tiger printer commands, the use of expansion cards, and a section on the Apple //e. Although some of the information in this book is dated, it is a convenient reference work dealing with Apple computers and some of the products that run on Apple computers. *Computer Station.*

Books* The following is a selection of books available on various topics relating to Apple computers and products:

Title	Vendor
Advanced Applesoft BASIC	MECC*
Advanced 6502 Interfacing	Group Technology Ltd.
Apple BASIC	Prentice-Hall
Apple BASIC for Business for the Apple II	Prentice-Hall
Apple BASIC: Data File Programming	John Wiley & Sons, Inc.
Apple II/IIe Computer Graphics	Brady Communications Co., Inc.
Apple Graphics Activities Handbook	Brady Communications Co., Inc.
Apple Graphics & Arcade Game Design	The Book Company
Apple Interfacing	Howard W. Sams & Co.
Apple Logo—Activities for Exploring Turtle Graphics	Brady Communications Co., Inc.
Apple Machine Language	Prentice-Hall
Apple Pascal	McGraw-Hill Book Co.
Apple Pascal User's Guide	MECC*
Apple Pascal Games	Sybex
Apple II Programming Exercises	John Wiley & Sons, Inc.
Apple II Quick Reference Guide	MECC*
Apple User's Encyclopedia	The Book Company
Apple II User's Guide, The	Osborne/McGraw-Hill
Applesoft BASIC for the Apple II/IIe	Brady Communications Co., Inc.
Applesoft Language	Howard W. Sams & Co.
Assembly Language Programming for the Apple BASIC User	Northern Technology Books
Basic Apple BASIC	Hayden Book Company

Title	Vendor
Basic BASIC—English Dictionary	Dilithium Press
BASIC Conversions Handbook, The	Hayden Book Company
BASIC Engineering, Scientific, and Business Programs for the Apple II/IIe	Brady Communications Co., Inc.
BASIC Programs for Scientists and Engineers on the Apple II	Brady Communications Co., Inc.
Beginning Applesoft BASIC	MECC*
Beneath Apple DOS	Quality Software
Circuit Design Problems for the Apple	Howard W. Sams & Co.
Circuit Design Programs for the Apple II	Group Technology Ltd.
Computer for Kids/Apple II + Edition.	Sensational Software
Computer Graphics Primer	Howard W. Sams & Co.
Computer Tutor for the Apple II, The	Prentice-Hall
Computer Tutor, The	Little Brown & Co.
Creative Apple, The	Creative Computing Press
Data File Programming for the Apple Computer	John Wiley & Sons, Inc.
Engineering Applications of Computer Graphics	Kern Publications
Everything You Wanted to Know About... Why People Use Apple Computers	Sterling Swift Publishing Co.
Free Programs for Your Apple	Sybex
Games for the Apple Computer	John Wiley & Sons, Inc.
Guide to Developing Instructional Software for the Apple II Micro-computer, A	MECC*
Graphic Software for Microcomputers	Kern Publications
Graphics Cookbook for the Apple Computer	Scelbi Publications
Handbook of Applesoft BASIC for the Apple II/IIe	Brady Communications Co., Inc.
Inside the Apple IIe	Brady Communications Co., Inc.
Instant BASIC—2nd Edition	Dilithium Press
Interface Projects for the Apple II	Prentice-Hall
Intermediate Applesoft BASIC	MECC*

Title	Vendor
Introduction to Applesoft BASIC—Student Text	MECC*
Introduction to the Apple II in Instruction	MECC*
Magic Turtle—Logo for the Apple IIc	Brady Communications Co., Inc.
MECC Book for the Apple, The	MECC*
Microbook: Database Management for the Apple II	Dilithium Press
Microcomputing Systems and Apple BASIC	Sterling Swift Publishing Co.
Microcomputer Workbook	Sterling Swift Publishing Co.
Mostly BASIC—Applications for Your Apple II	Howard W. Sams & Co.
Pascal Programming on the Apple	Prentice-Hall
Practical BASIC Programs—Apple II	Osborne/McGraw-Hill
Programmer's Handbook for the Apple II	Computer Station
Programming the Apple	MESA Research
Programming the Apple, A Structured Approach, Revised and Enlarged	Brady Communications Co., Inc.
Programming Tips and Techniques for the Apple II/IIe	Brady Communications Co., Inc.
Science & Engineering Programs—Apple II Edition	Osborne/McGraw-Hill
Some Common BASIC Problems—Apple II	Osborne/McGraw-Hill
Training Your Computer—Apple Edition	A. R. Davis & Co.
Turtlesteps—An Introduction to Apple Logo	Brady Communications Co., Inc.
Using the Computer in the Classroom (Apple Version)	MECC*
Using 6502 Assembly Language—How Anyone Can Program the Apple II	Datamost, Inc.
What's Where in the Apple?	Micro Ink
32 BASIC Programs for the Apple Computer	Dilithium Press

* Minnesota Educational Computing Consortium.

MAGAZINES

A+ Called the "Independent Guide for Apple Computing", this monthly magazine includes articles, reviews, and evaluations of products and items of interest to Apple owners. It is published by:

Ziff-Davis Publishing Company.
One Park Ave.
New York, NY 10016

A+ Disk Magazine A magazine of useful computer programs for Apple computers. Each disk includes up to 12 programs with a complete User's Guide for each. Most of the programs run under DOS 3.3. One year's subscription consists of six disks. Published by Ziff-Davis Publishing Co.

Apple Orchard The *Apple Orchard* is a monthly magazine containing articles, reviews, and some tutorials on items of interest to Apple owners. Also included is information about Apple user groups and new products. It is published by:

International Apple Core
908 George Street
Santa Clara, CA 95050

Apple Tech Notes *Apple Tech Notes* presently includes more than 700 pages of information on hardware, software, and internal parts of the Apple, with quarterly updates available by subscription. The initial purchase gives you a binder and the current set of pages. *International Apple Core.*

inCider This magazine includes departments on Applesoft tutorials, software and book reviews, hints and techniques, and articles of all types, from reviews to humor to discussions of where Apple will go from here. *inCider* is a monthly publication. The address is:

inCider Subscription Services
P.O. Box 911
Farmingdale, NY 11737

Infoworld *Infoworld* is a weekly newspaper about microcomputers. Valuable for keeping up with new developments on a week-by-week basis. Most computer stores and larger bookstores carry *Infoworld.* *Infoworld's* address is:

Infoworld
Circulation Dept.
Box 837
Framingham, MA 91701

Microcomputing This is a monthly general interest computer magazine, geared to the computing novice. Published by:

Wayne Green, Inc.
Box 981
Farmingdale, NY 11737

Nibble Considered by many, including the authors, to be THE magazine for the Apple computer. *Nibble* emphasizes program listings, which you can key in yourself. These programs range from the business-oriented to games to outright craziness (watch out for anything written by Paul Raymer; if you have just had an operation, you could burst your stitches laughing!). The programs are written in a combination of Applesoft and machine language. There are directions for entering the machine code. If you prefer, you may order a disk with the principal programs of each issue. *Nibble* also publishes *Nibble Express*, which binds the major articles and programs for a year's (8) issues, including with the discussion any upgrades and corrections which have been discovered since the original publication. The subscription address is:

Nibble
MicroSparc Inc.
Box 325
Lincoln, MA 01773

Peelings II A magazine of Apple software and hardware evaluations. Includes tutorials, glossaries, and comparision charts on the most popular and the newest products for the Apple family of computers. A one-year subscription consists of 9 issues. Published by:

Peelings II, Inc.
Box 625
Holmes, PA 19043

Softalk This monthly publication contains extensive listings of software, monthly "top ten" and "top thirty" listings, articles about people in the industry, and several monthly columns, including those dealing with assembly language, BASIC, Apple //c, and ProDOS. Also included is a monthly new product information section and reviews. Published by:

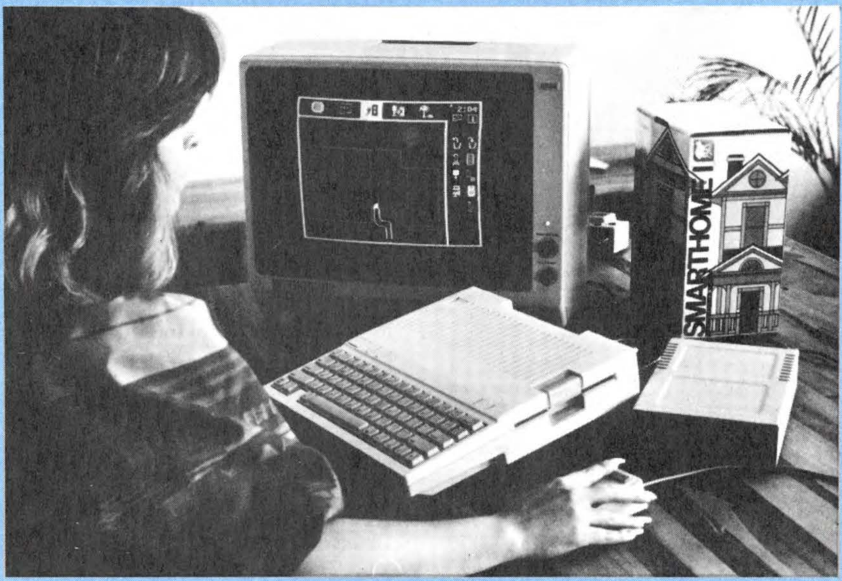
Softalk Publishing, Inc.
Box 60
North Hollywood, CA 91603

SoftSide *SoftSide* publishes articles of general interest applying to several different computers. Depending upon your request at the time of subscribing, they include a "tear-out" section with two to

four program listings per month especially for your machine. They also produce a disk version and a cassette version of the magazine with much more expanded programs, but also for a considerably higher price. The use of "tear-out" inserts does not allow direct comparison of program listings for different computers (you may order the tear-outs separately for other computers if you wish). The subscription address is:

SoftSide Publications
100 Pine Street
Holmes, PA 19043

Appendix B
User Groups



User Group A user group is a group or club focused on an aspect of computers. Some clubs direct their attention to one particular computer; there are many Apple Users Groups, some of which are listed below. Other groups focus on a language (FORTH Users Group, Pascal Users Group, and others), on an Operating System (CP/M, and others), on an area of application (accounting, education, science, graphics, and others), or other aspects of computing. Many magazines on computing (q.v.) carry lists of clubs/groups and report on their activities.

User groups provide a valuable opportunity to get and give advice on Apple hardware, software, and applications. Often you can talk to someone who used a product you are considering buying. User group newsletters may also offer useful information. If you need a programmer or consultant, you may meet or hear about a good one at a user group meeting.

ALABAMA

Applebamians
Thomas Moore
P.O. Box 1588
Anniston, AL 35630

Apple Corps of Birmingham
Roland Turrentine
1037 F Huffman Rd.
Birmingham, AL 35215

West Alabama Users
Barry Collins, President
1009 W. Jackson St.
Demopolis, AL 36732

Quad Cities Apple Byters
Leslie R. Tate
Rt. 6 Box 304
Florence, AL 35631

Coosa Valley Apple Club
Charles Stanton Jr.
226 South 5th St.
Gadsden, AL 35901

Newton's Tree Apple Group
Geoffrey Hintle, Secretary
1500 Sparkman Dr., Apt. 38F
Huntsville, AL 35805

SAPPLE
Jerry Broadhurst
1224 W. Vendom Dr.
Mobile, AL 36609

Shelby Core
Nathan Meminn
P.O. Box 332
Montevallo, AL 35115

Montgomery Apple Users
Chuck Wildzunas
649 Williamson Rd.
Montgomery, AL 36109

Apple M.U.G.
Gary Marston
2841 Shenandoah Dr.
P.O. Box 20241
Montgomery, AL 36116

Peanuts and Apples
Jack Cumbie
Rt. 2, P.O. Box 50
Ozark, AL 36360

ALASKA

Anchorage Apple
Dennis McWilliams
2619 Glenkerry Dr.
Anchorage, AK 99504

Apple Mousse
John Franich
SR Box 30320-G
Fairbanks, AK 99701

ARIZONA

Jeddah Computer Club
Robert McGowen
218 N. 58th St.
Mesa, AZ 85205

Apple C.A.R.T.
Warren Michelsen, President
P.O. Box 2361
Page, AZ 86040

Adam II
Ron LaMee
2808 E. Larkspur Dr.
Phoenix, AZ 85032

Gila Valley Apple Assn.
Don Lancaster
746 1st St.
P.O. Box 1300
Thatcher, AZ 85552

Tucson Apple Use
James Maginnis
1604-C East Blacklidge Dr.
Tucson, AZ 85719

Yuma Apple Users
Willard Robinson
1900 Paris Ct.
Yuma, AZ 85364

ARKANSAS

Razorback H.A.U.G.
Robert P. Hughes
3000 Rogers Ave.
Fort Smith, AR 72901

Little Rock Apple Addicts
8500 S. University
Little Rock, AR 72209

AMERICAN SAMOA

S.P.I.C.E.
John Flanigan
P.O. Box 1087
Pago Pago, AS 96799

CALIFORNIA

M. Kent Hockabout
HBI Computer Group
936 Shorepoint Ct.
Alameda, CA 94501

S.A.K.W.A
MSG Sanford Greenfarb
HQ USFK/EUSA-J2
APO San Francisco, CA 96301

Tabacheck User Group
PSC #4 Box 17363
APO San Francisco, CA 96408

Misawa Computer Club
James Story
PSC Box 6002
APO San Francisco, CA 96519

Bakersfield Apple
R. A. Campbell
3005 Horace Mann Ave.
Bakersfield, CA 93306

A-Bug
Virginia Bidgood
90 W. HSY 246
P.O. Box 1985
Buellton, CA 93427

Lerc Aces
Rod White, Secretary
P.O. Box 551
Burbank, CA 91520

Tri-Network Apple Group
Jeff Mazur, President
8041 Sadring
Canoga Park, CA 91304

North County Computer Club
William Sugg
2010 Pintoresco Ct.
Carlsbad, CA 92008

Hesea Apple Use
Bud Grove, President
21111 Dolores, #146
Carson, CA 90745

Apple Sac
Marge Jurach
6630 Willowleaf Dr.
Citrus Heights, CA 95610

Apple Corps of San Diego
William Rupp
588 W. Chase Ave.
El Cajon, CA 92020

Apple Mug
Marilyn Melhorn
865 Saint George Dr.
El Cajon, CA 92021

Orange Apple Computer Club
Stephen P. Blythe
22501 Eloise Dr.
El Toro, CA 92630

ITTCE Apple Users
Brad Ford
10550 Talbert Ave. T-105
Fountain Valley, CA 92708

McDonnell Douglas Astro Group
Larry Teselle, President
16681 Mt. Darwin Circle
Fountain Valley, CA 92708

SPCA
Jim Weil, President
4781 Calaveras Ave.
Fremont, CA 94538

Sierra Apple Orchard
Bruce Riley
6506 N. Kennedy
Fresno, CA 93710

HFEA Apple Computer Group
Don Ardert
424 W. Commonwealth
Fullerton, CA 92632

North Orange Computer Club
Bob Brown
11121 Vinevale
Garden Grove, CA 92641

Sequoia Computer Users
Victor Coelho
216 East Colonial Dr.
Hanford, CA 93277

Arabian Apple Assoc.
H. Hollerman DH-928
10834 S. La Cienega
Inglewood, CA 90304

Ridgecrest Apple Group
Gene Thomas, President
Star Route Box 109E
Inyokem, CA 93527

JPL Apple Club
Arthur Collins, President
4238 Wiley Ln.
La Cresenta, CA 91214

Desert Apple Computer
Tom Meier, President
P.O. Box 2702
Lancaster, CA 93539

Original Apple Corps
Elroy Wieck
4633 W. Compton Blvd., #152
Lawndale, CA 90260

Lodi Apple Club
Joan Pipes, Business Manager
816 W. Lockeford St.
Lodi, CA 95240

L.A. Apple Users Group
Philip Wasson, President
9513 Hindry Pl.
Los Angeles, CA 90045

TRWAUG
Kim Schuette
7530 West 80th
Los Angeles, CA 90054

S.P.A.C.E.
Roger Chaffee
1030 Ringwood Ave.
Menlo Park, CA 94025

SRI Int'l Apple Club
Steven Leach
333 Ravenwood Ave.
Menlo Park, CA 94025

Stanislaus Apple Group
George Osner
1124 Brady Ave.
Modesto, CA 95350

Ames Computer Club
Att: James Cox
MS 243-5
Moffett Field, CA 94035

Apple Electrosystems
John Bowers/Aerojet
533 Broadoaks Ave.
Monrovia, CA 91061

Silicon Apple PMG SOC
Lowell Noble
18138 Bancroft Ave.
Monte Sereno, CA 95030

Naval Postgraduate School Computer Club
Phillips Johnson
105 Malloway Ln.
Monterey, CA 93940

ABACUS
Jim McMonagle
23 Buckingham Dr.
Moraga, CA 94556

CSUN Apple Users
Computer Science Dept.
18111 Nordhoff St.
Northridge, CA 91330

Apple Pickers II
John Balshaw
1680 Kearny Ct.
Petaluma, CA 94952

TVAUG
Bill Smith
3525 Kings Canyon Ct.
Pleasanton, CA 94566

South Bay Apple Club
Jerry Parker
27906 Indian Rock Dr.
Rancho Palos Verdes, CA 90274

T.R.A.C.E.
P.O. Box 6007-169
1689 Market St.
Redding, CA 96001

Cubic Corp Apple Sig
Jeffrey Stumpf
9333 Balboa Ave. m/s 10-31
San Diego, CA 92021

San Diego Medical Group
Joel Kunin
7920 Frost St., Suite 405
San Diego, CA 92123

Convair Association Computer Club
Mark Jones
P.O. Box 80847 MZ54-116
San Diego, CA 92138

San Francisco Apple Core
1515 Sloat Blvd., Suite 2
San Francisco, CA 94132

UCSF Apple Users Group
Billy Bargagliotti
University of California 345
San Francisco, CA 94143

Blossum Valley Apple Club
Ernest Webb, President
5821 Cottle Rd.
San Jose, CA 95123

Apple San Luis Users Group
Robin Grinnell, President
553 Couper Dr.
San Luis Obispo, CA 93401

Logicon Microcomputer
Tom Wallace
255 W. Fifth St.
San Pedro, CA 90733

Santa Barbara Apple Group
3313A State St.
Santa Barbara, CA 93105

Apple P.I.E.
Bill Nienhaus, President
337 Montclair
Santa Clara, CA 95051

SAUG
Joel Amromin, Treasurer
2633 10th St., #2
Santa Monica, CA 90405

Apple - SMUG
Marilyn Dresbach
780 Trinity Ave.
Seaside, CA 93955

Apple Peelers
Gene Wilson, President
17126A Cedar Ave.
Sonoma, CA 95476

Lockheed Apple Group
Robert B. Kittle
780 Ponderosa Ave.
Sunnyvale, CA 94086

Rancho California Computer Club
Joe Dorner
P.O. Box 651
Temecula, CA 92390

Midway Computer Club
Tony Severa
131 Highland Ave.
Vacaville, CA 95688

Pyrus Malus
Monty Kosma
25681 Estoril St.
Valencia, CA 91355

S.M.A.L. Apple
Ronald Stankiewicz
1114 Cypress
Vandenberg AFB, CA 93437

Appleholics Anonymous
Roger Lucic
3875 Telegraph Road
Ventura, CA 93003

Apple for the Teacher
David Miller
9525 Lucerne
Ventura, CA 93004

Honeywell 6502 Computer
Lew Earer/MS-444A
1200 E. San Bernardino Road
West Couina, CA 91790

Apple PI
David Kay
1033 Stoneshead Ct.
Westlake Village, CA 91361

COLORADO

Honeywell Apple Users
Sue Mallinger
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906

Apple PI/Denver
The Seed
P.O. Box 17467
Denver, CO 80217

Rocky Apple Core
Cyrus Wood, President
730 String St.
Estes Park, CO 80517

South Colorado Apple Users
Tom Thomas, President
1635 S. Prairie
Pueblo, CO 81005

CONNECTICUT

Danbury Apple Users
Bill Morelli M/S 807
100 Wooster Hts. Road
Danbury, CT 06810

Applefield Users Group
Marc Lillie, Vice President
1700 Post Rd.
Fairfield, CT 06430

Southern Eastern Connecticut Apple
Warren E. Nordgren
42 Terry Rd.
Gales Ferry, CT 06335

Applelist
Willis A. Hunt, President
156 South Ave.
North Haven, CT 06473

Apple Mugs
Jim Courte, President
354 Main Ave.
Norwalk, CT 06851

Huge Apple Club
Richard Johnston
29 Ridge Road
Simsbury, CT 06070

SNAP
Ray Badowski
75 Rustown Dr.
Stratford, CT 06497

Appleshare
Roman Juskiewicz, President
P.O. Box 696
Westport, CT 06881

DELAWARE

GRAPE
P.O. Box 8904
Newark, DE 19711

DISTRICT OF COLUMBIA

Nairobi Apple Users Group
Walter Hecox, RMEA Nairobi c/o
Mail Rm., World Bank, 1818 H St. NW
Washington, DC 20433

Gerald V. Van Der Vlugt
Kathmandu (ID)
Dept. of State
Washington, DC 20520

FLORIDA

Tom Zornes
Richard Fisher
PSC Box 2166 or 1032
APO Miami, FL 34002

Capeapple
John McNally
1661 Crooked Arrow Ct.
Cape Coral, FL 33904

Fac-Net
Hilliard Jason
1320 Dixie Hwy., Rm. 900A
Coral Gables, FL 33146

Polk Apple Core
954 E. Lexington St.
Lakeland, FL 33801

Apple Tree of Central Florida
Bernice Andrews
2341 Carolton Rd.
Maitland, FL 32751

Space Coast Apple Group
Elmer L. Whalley, Treasurer
110 Carib
Merritt Island, FL 32952

Miami Apple Users Club
Paul Gargone, President
5190 N.W. 167 St.
Miami, FL 33014

Dade Radio Computer Club
Leo Toussaint
7500 S.W. 82nd Ave.
Miami, FL 33143

Smaug
Phil Mitchell
10201 Fontainebleau Blvd., #206
Miami, FL 33172

A.C.E.S.
Marty Martin
6902 Rio Pinar
N. Lauderdale, FL 33068

Sun Coast Computer Assoc.
A. C. Wall
705 Bayshore Rd.
Nokomis, FL 33555

Daytona Beach Apple
Bob McFarland
30 Arrowhead Circle
Ormond Beach, FL 32074

G.S.I. Computer Club
1165 E. Blue Heron
Singer Island, FL 33404

Honeywell Apple Users
Bill Isreal
2015 Dolphin Blvd. St.
St. Petersberg, FL 33707

Scat
Earl Greene, Jr., President
4140 Bayshore Blvd. NE
St. Petersberg, FL 33703

Suncoast Tampa Apple
STAG
7604 Sanibel Circle N
Tampa, FL 33617

Venice Apple
Derek Dunn-Rankin
200 E. Miami Ave.
Venice, FL 33595

GEORGIA

Classic Apple Association
W. M. Peters
407 Meadow Lane
Athens, GA 30605

Mite
Jack Butler
2795 Lexington Rd.
Athens, GA 30605

AMIGOS
Susan Jordon
106 Anderson Ave./P.O. Box 301
Fort Valley, GA 31030

Widget Apple Group
R. D. Henning
3396 Timbercreek Dr.
Lawrenceville, GA 30245

HAWAII

H.A.U.S.
Ron Letson
P.O. Box 91
Honolulu, HI 96810

M.A.U.I.
Gary Wehe, President
37 Malalani Place
Makawao, HI 96768

IDAHO

A.B.U.G.
Merle Lippott
3933 Constitution Way
Boise, ID 83706

ILLINOIS

NIAUG
1271 West Dundee Rd.
Buffalo Grove, IL 60090

Centralia Apple Users Group
John Laughhunn
418 S. Poplar
Centralia, IL 62801

Crabapples
John P. Russell
Rt. 2, Box 197
Cobden, IL 62920

Apple People Users Group
Donald Rose
224 N. Main St./P.O. Box 333
Crystal Lake, IL 60014

Danville Area Apple Compatible User's
Dave Nicholson
23 Dixie Acres Rd.
Danville, IL 61832

C.I.A.
Dan and Marge Kissling
118 Ritchie
E. Peoria, IL 62401

Mid-Illinois Apple Users
Charles Vonbokel
Box 526
Effingham, IL 62401

Plato Center Apple Group
William Swanson
Rt. 2, Box 189
Elgin, IL 60120

S.H.A.G
Lawrence Carr
1308 Cumberland Circle E.
Elk Grove, IL 60007

Apple Users
Gustave Rath, Indusrial Engineering
Northwestern University, Rm 1761
Evanston, IL 60201

The Apple Tree
Dale Novak
18735 Crawford
Flossmoor, IL 60422

Metro East Computer Association
Wilbur Felkey
5238 Englewood
Godfrey, IL 62035

Dupage Co. Apple Users
Earl Allen
1241 Citation Lane
Hanover Park, IL 60103

Oily Apples
Larry Wisinski, President
400 E. Sibley Blvd
Harvey, IL 60426

Ideal Apples
Robert Chinn
101 S. Schuyler Ave.
Kankakee, IL

Palatine High School
Christine Person
1111 North Rohlwing Dr.
Palatine, IL 60067

Quincy Apple Club
Larry Schaffer
4729 Broadway
Quincy, IL 62301

D.A.T.A.
Joe Travis, President
2709 Silent Wood Trail, #8
Rockford, IL 61109

Medico-Legal Apple Club
Robert Book
509 Shorewood
Shorewood, IL 60435

Capital Apples
John Venvertloh
301 Saxon Dr.
Springfield, IL 62704

Champaign-Urbana Apple Group
Sidney Bowhill, President
2203 Anderson
Urbana, IL 61801

Apple Pie of Illinois
Jack Gratz, President
11630 S. Nagle Ave.
Worth, IL 60482

INDIANA

The Apple Pickers, Inc.
c/o John Haver
1300 Albany Ave.
Beech Grove, IN 46107

Caug
Charles Hatcher, President
2805 Chestnut Ct.
Columbus, IN 47201

Fort Wayne Apple Group
Joe Kucharski, President
P.O. Box 11507
Fort Wayne, IN 46858

Apple Tech
Michael Miller, President
1755 Linden Ave.
Mishawaka, IN 46544

North Liberty Computer
Robert Brown
25352 Martindale Rd.
North Liberty, IN 46554

Wabash Valley Apple Byters
Bernard Ridens
RR 21, Box 191
Terre Haute, IN 47802

Tri-State Apple
Robert Collins
P.O. Box 599
Vincennes, IN 47591

IOWA

Apple Micro Enthusiasts Society
Terry Henderson
213 Lincoln Way
Ames, IA 50010

Bedford Comm Apple Users
Neil Hall
908 Penn
Bedford, IA 50833

Green Apples
John Buttrey
3412 Clover Hills Dr.
Bettendorf, IA 52722

Green Valley Users Group
Dave Blair
Green Valley Area 14
Creston, IA 50801

Apple Byters
Scott Lee, President
1719 Wilkes Ave.
Davenport, IA 52804

The Green Apples
Larry Snyder
6239 Eldora Rd/Box 544
Hudsen, IA 50643

Iowa User Group
Joann Short
844 10th N.E.
Mason City, IA 50401

Rolfe Apple Club
Lee Thorson
RR 1
Rolfe, IA 50581

Agri-Cursors
Neil Stadlman
500 Audobon
Sac City, IA 50583

Sioux-Apple
A. Kleider
28 West Kings Highway
Sioux City, IA 51104

B.C.C.
T. Kleen/Briar College
3303 Rebecca St.
Sioux City, IA 51104

KANSAS

Kechi Computing
328 Cherokee Dr./ P.O. Box 160
Kechi, KS 67067

South Kansas City Apple User Group
11915 Overbrook Rd.
Leawood, KS 66209

Postrock Apple Club
Vernon Snider
101 E. Lost
Lincoln, KS 67455

Apple Bits
Robert Mills
6140 Glenwood
Mission, KS 66202

N.W. Kansas Prairie Apples
Mark Ward
Box 156, 101 S. Decatur Ave.
Norcatur, KS 67653

Plane Apple Club
John Van Walleghen
Box 12013
Wichita, KS 67277

KENTUCKY

Lower Cumberland Green Apple
Major James Yule
1038 Drennan
Ft. Campbell, KY 42223

Southern Indiana Apple Club
R. Browning, Sr.
1019 Fernwood Dr.
Henderson, KY 42420

Louisville Apple Group
Linda Lewis
2722 Rodman St.
Louisville, KY 40217

Paduke Apple Users Group
Gemological Serv., Suite 620
Citizen's Bank Building
Paducah, KY 42001

LOUISIANA

Apple Users of Alexandria
Jim Hurst
3208 Pine Ridge Dr.
Pineville, LA 71360

Northeast La Apple Club
Harper Gaushell
2001 N. 7th St./P.O. Box 1829
West Monroe, LA 71291

MAINE

Southern Maine Apple Users
Harkins
Pleasant Hill Rd.
Freeport, ME 04032

MARYLAND

Baltimore Apple Users Group
Abel Merrill
P.O. Box 13522
Baltimore, MD 21203

Maryland Apple Corps
Art Blumberg, Secretary
6711 Greenspring Ave.
Baltimore, MD 21209

Washington Apple Pi
David Morganstein, President
8227 Woodmont Ave.
Bethesda, MD 20814

Maryland Crabapples
Logical Choice
Chatham Mall
Ellicott City, MD 21043

Pencom
John Pence, President
222 Key Parkway
Frederick, MD 21701

Frederick Apple Core
Randy Pasley
1835 Meadowgrove Lane
Frederick, MD 21701

Washington Apple Pie
13329 Woodruff
Germantown, MD 20874

APL Apple Users Grp.
Edward Davis, Secretary
Johns Hopkins Applied Physics
Laurel, MD 20707

E. Shore Crab Apple Club
Harvey Neddleman
Valleywood Dr., Rt. #7/Box 31A
Salisbury, MD 21801

MASSACHUSETTS

Mitre Apple Society
c/o Balwim Troutman
11 Norma Rd.
Bedford, MA 01730

Apple Boston
Boston Computer Society
3 Center Plaza
Boston, MA 02108

Applesauce
Todd Marchant
130 Meadow Rd.
East Longmeadow, MA 01028

Bristol County Computer Club
Ernest P. Viveiros
945 Walnut St.
Fall River, MA 02720

Appleseed
Al Berkowitz
1550 Worcester Rd., #321
Framingham, MA 01701

Apple Core Berkshire Co.
H. F. Allan
14 Hutchinson Lane
Lenox, MA 01246

Newton Apple Kids
Raines Cohen
50 Court St.
Newtonville, MA 02160

North Shore Apple Group
Robert Ramsdell, President
Panxsphics Whistlestop
Rockport, MA 01966

Ray Apples Users Group
H. Tate
20 Normandy Dr.
Sudbury, MA 01776

Cape Cod Apple User Group
Robert Baker
2 High Grove Rd.
Yarmouth, MA 02664

MICHIGAN

CPAC
Scott Goss
124 E. 4th St.
Flint, MI 48502

Apple LUG
Dale Marshall
5411 Marsh Rd.
Haslett, MI 48840

Apple Angels
Milton J. Kessie
16820 Hubbard Rd.
Livonia, MI 48154

Michigan Apple Club
John Zellji
P.O. Box 551
Madison Heights, MI 48071

Grand Rapids Apple
Ted Bailey
3311 Flamingo S.W.
Wyoming, MI 49509

Ann Arbor Apple
Steve Opal, Editor
3716 Gulf Side Dr.
Ypsilanti, MI 48197

MINNESOTA

Anoka Computer Club
Don Venne
229 Jackson St., #110
Anoka, MN 55303

Bemidji's Ongoing Computer
KAWE-TV D Checots
BSU, Box 9
Bemidji, MN 56601

Lake Superior Apple User Group
John Buchanan
16 W. First St.
Duluth, MN 55802

Mini/App/les
Ron Androff
1725 Crest Ridge Lane
Eagan, MN 55122

The Computer Club
Bill Fredrich
230 N.E. 2nd St.
Faribault, MN 55021

Mayo Apple Users Group
A. E. Pertuz, Treasurer
Sec. of Sys. & Proc., Mayo Clinic
Rochester, MN 55901

Personal Computer Interest Group
Recreational Services
3606 Hwy 52 North
Rochester, MN 55901

Woodbury Computer Club
Dan Kirby
1750-21 Weir Dr.
Woodbury, MN 55125

MISSOURI

Personal CMP - St Louis
Ben Atwater
15672 Silverlake Ct.
Chesterfield, MO 63017

A.M.M.P.L.E.
Century West
1001 E. Walnut
Columbia, MO 65201

Apple Eye
Michael Rumelt, MD, President
10914 Brook Trail Ct.
Creve Coeur, MO 63141

Joplin Apple Users Group
Ross Lauck, President
1903 E. 36th
Joplin, MO 64801

Apple Squires of the Ozarks
1904 E. Meadowmere
Springfield, MO 65804

Apple Jacks
1383 White Orchard
St. Louis, MO 63146

Comstat
11610 Page Services Rd.
St. Louis, MO 63141

MISSISSIPPI

H.A.U.G.
Francis Farmer
1402 Mamie St.
Hattiesburg, MS 39401

Meridian Computer Club
Att: Bob Walters
3304 20th St.
Meridian, MS 39301

Natchez Apple User Group
Paul Fletcher
21 Woodside Ct.
Natchez, MS 39120

MONTANA

Apple Addicts
Wiltse
Box 489
Broadus, MT 59317

Big Sky Apple
Bob Hawley
3400 5th Ave. South
Great Falls, MT 59405

Apple SOS
Box 2051
Wolf Point, MT 59201

NEBRASKA

Compusers
Dorothy Friend
219 N. Burlington Ave.
Hastings, NE 68901

Apple-Linc
Bruce Baasch, Treasurer
334 North 70th
Lincoln, NE 68505

Big Red Apple Club
John Wrenholt, President
1301 N. 19th
Norfolk, NE 68701

Omaha Apple Suace
John Anderson
3103 S. 32nd Ave.
Omaha, NE 68105

NEW HAMPSHIRE

Suffolk Apple CMP SOC
M. Weinstock
226 Boston Post Rd.
Amherst, NH 03031

Big Green Apple
Norm Chapman
1 Ledge Rd.
Hanover, NH 03755

Labcom Users Group
John Lawe
Mary Hitchcock Memorial Hospital
Hanover, NH 03756

Southern N.H. Apple Core
George H. Abbott
318 Central St.
Hudson, NH 03051

Pemi-Baker Computer Group
Henty Ahern
RFD #2, Box 339
Plymouth, NH 03264

NEW JERSEY

Monmouth Apple Corps
H. C. Logan, Jr., President
193 Stillwells Corner Rd.
Freehold, NJ 07728

Apple of The Eye
Stephen Mendelsohn
64 Maiden Lane
Little Ferry, NJ 07643

Apple Jack
John Parker
40 Green Ave.
Madison, NJ 07940

Atlantic County Computer Club
William A. Merlino
92-B Somers Point Rd.
Mays Landing, NJ 08330

The Apple Synapse II
David Bostain, President
1050 George St. 5-0
New Brunswick, NJ 08901

Southern NJ Apple Group
Phil De Vicci
3612 Remington Ave.
Pennsauken, NJ 08110

Amateur Computer Club
Seymour Joseph
5 Bayberry Close
Piscataway, NJ 08854

Princeton Apple Group
P.O. Box 431
Rocky Hill, NJ 08553

Short Hills Apple Pits
Steve Block, President
29 Clive Hills Rd.
Short Hills, NJ 07078

Ewing Apple Users Group
Bob Austin
16 Riverview Dr.
West Trenton, NJ 08626

NEW MEXICO

Applequerque Computer Club
Jame Segrest
6609 Othelia Ave. N.E.
Albuquerque, NM 87109

R.A.F. Computer Club
Dennis Quillen
4108 Douglas
Farmington, NM 87401

Santa Fe Computer Club
Dick Laine
2801 Calle Campeon
Santa Fe, NM 87501

NEW YORK

Bad Appleing Computer
Eden Recor
Unit AA, Box 982
APO New York, NY 09098

Macro
Elias Ghuneim
37 Alhambra Rd.
Baldwin, NY 11510

Brooklyn Friends School Apple Club
Jerome Vuoso
164 Neck Rd.
Brooklyn, NY 11223

Apple Byters Corps, Inc.
171 Treehaven Dr.
Buffalo, NY 14215

Orange County Computer
Walter Shave, Jr.
RR 1, Box 34
Campbell Hall, NY 10916

Apple C.I.D.E.R.
E. Titlebaum, President
674 Quaker Meeting House Rd.
Honeoye Falls, NY 14472

Big Apple Users Group
John McMullen, President
Perry Street
Jefferson Valley, NY 10535

Apple Power
Ken Davis, Vice President
26 Peter Lane
Plainview, NY 11803

Adirondack Apple Core
Raney Ellis
82 Court St.
Plattsburg, NY 12901

Staten Island Apple
Mr. T. Baker
100 Piave Ave.
Staten Island, NY 10305

TSAUG
Thomas Coulter
P.O. Box 6586, Teall Avenue Station
Syracuse, NY 13217

Mid Hudson Micro Users
Dale Alexander
Imperial Plaza 118, Rt. 9
Wappingers Falls, NY 12590

West Hempstead Apple
Lance Modell, President
400 Nassau Blvd.
West Hempstead, NY 11552

Crab Apple
Harry Geist, President
18 Hobe St.
West Nyack, NY 10994

U.A.U.G. % Upstate Com
RD #1, Box 17-A99 Commercial Dr., Rt 5A
Whitesboro, NY 13492

NORTH CAROLINA

Green Apples
Nancy Terrell, President
2604 Battleground
Greensborough, NC 27408

Hillsborough Orchard Surv.
Bill Boyarsky, President
Rt. 1, Box 242 E
Hillborough, NC 27278

Carolina Apple Core
Alec Whittaker, President
P.O. Box 31424
Raleigh, NC 27622

NC State Apple Users
English Research Division
P.O. Box 5995
Raleigh, NC 27650

N.C.E.C.S. Apple Users Group
David Hinnant/STRC Bldg.
Corner Davis Dr. & Cornwallis Rd.
Research Triangle Pk, NC 27709

Unc-ch Apple Group
Kathryn Conway NC Media Center
07 Smith Bldg., 128A
Chapel Hill, NC 27514

Charlotte Apple Computer Club
James Eddings
1218 Jones St.
Gastonia, NC 28052

NORTH DAKOTA

G.F. Apple S.A.U.C.E.
Paul Kobe/Computerland
2500 B South Columbia Rd.
Grand Forks, ND 58201

Apple Polishers
Cpt. William D. Stromire
1045 W. Central Ave.
Minot, ND 58701

Farmers Computer Club
Leon Kirschemann
RR #2
Regent, ND 58650

OHIO

Neo Apple Corps
Tom Wysocki
7047 East Jefferson Dr.
Mentor, OH 44060

Akron Apple Association
Richard L. Meyers
679 S. Cleveland Ave.
Mogadore, OH 44260

The Country Computer Club
Elec. Cntr/N. Russell
8769 Twonsinfo, RD 513
Shreve, OH 44676

Mahoning Valley Apple
Gary Bennett
5066 Mahoning Ave. N.W.
Warren, OH 44483

C.O.A.C.H.
Travis Risner
193 Hamilton Ave.
Westerville, OH 43081

Apple - Dayton
Gary Adams
101 Dupont Way
WP AFB, OH 45433

ASCC
Chuck Savage
1175 Linden Ave.
Akron, OH 44310

Rubber Apple Users Group
Robert Nebiker, President
1870 Brookfield Dr.
Akron, OH 44313

Apple Friendly
Robert Hartman
9680 Crackel Rd.
Aurora, OH 44202

Toledo Apple Users
Carl Nevardo
14261 Georgetown Dr.
Bowling Green, OH 43402

Apple-Ciders
Cindy McDaniels, President
5001 Relleum Ave.
Cincinnati, OH 45238

Wendy's Int'l, Inc.
Thomas Mulcahy
4288 West Dublin-Granville Rd.
Dublin, OH 43017

OKLAHOMA

Edmond Apple Users
Dr. Barbara Ryan
12708 Teakwood
Edmond, OK 73034

S.W. Oklahoma Computer Club
Joe Teeples
3411 N.W. Floyd
Lawton, OK 73505

OKC Apple Users Group
Glenn Rodkin
3728 North Frankford
Oklahoma City, OK 73112

Pioneer Apple Club
Charlie Gawthrop
2105 Kyme
Ponca City, OK 74604

Tulsa Computer Society - Apple
Randy Chevrier
5950 E. 11th St.
Tulsa, OK 74112

OREGON

AARG
Charles Ryberg-Humanities
Southern Oregon State College
Ashland, OR 97520

Rouge Apple Users Group
David L. Burlingame
2020 Hyw. 99 N., #17
Ashland, OR 97520

Corvallis Apple Club
Jack Trowbridge, President
101 N.W. 23rd
Corvallis, OR 97330

Apple Blossom
Walter Leja, President
170 N. Prairie St.
Dillard, OR 97432

Appleugene
Charles Rush, President
1430 E. 27th St.
Eugene, OR 97520

Salem Area Computer Club
Ed Sigurdson
1115 Madison St. N.E., #601
Salem, OR 97303

Tillamook Computer
Bell's Office Machines
1906 Third St.
Tillamook, OR 97141

PENNSYLVANIA

Delaware Valley Apple Branch
Chris Hardy
32 Union Ave.
Bala Cynwyd, PA 19004

Apple of North Hills
Samuel Bell/Computerland
1378 Freeport Rd.
Blawnox, PA 15238

Sperry Computer Club
H. Lipschultz
Township Line & Jolly Rd.
Blue Bell, PA 19424

Chester County Computer Club
John Rosenberg
1107 Wynnbrook Rd.
Clifton Heights, PA 19018

Erie Apple Crunchers
Rudy Guy
1114 West 22nd St.
Erie, PA 16502

Indiana Area Apple User Group
Brian McCue
1680 Warren Rd.
Indiana, PA 15701

Keystone Apple Core
David Murdoch
4644 Carlisle Pike
Mechanicsburg, PA 17055

Monroeville Apple Users Group
Dr. Hsieh
3821 Harwick Dr.
Murrysville, PA 15668

PACS Apple User Group
Steve Longo, Physics Department
La Salle College
Philadelphia, PA 19141

Apple Pittsburgh
Alex de Poutiloff
1089 Greenlawn Dr.
Pittsburgh, PA 15220

South Hills Computer Club
Max Janairo
1145 Sunrise Dr.
Pittsburgh, PA 15243

Bac Berks Apple Club
Phillip Fraley, Treasurer
1753 York Rd.
Reading, PA 19610

Pennsylvania Apples
H. A. McKinstry
522 S. Pugh St.
State College, PA 16801

G. Pirrung
NADC
Code 3021
Warminster, PA 18974

RHODE ISLAND

Rhode Island Apple Core
Russell Ferland
126 Slater Park Ave.
Pawtucket, RI 02861

Apple Fritters
Jim Dugas
Butler Hospital, 345 Blackstone
Providence, RI 02906

SOUTH CAROLINA

Augusta Apple Users Group
A. S. Jennings
1469 Canterbury Place S.E.
Aiken, SC 29801

Lowcountry Apple Corps.
Computer Source
1660 San Rittenberg Blvd.
Charleston, SC 29407

CSC Apple Users Group
Rick Wrigley
1106 Haven Dr.
Columbia, SC 29209

TENNESSEE

Tri-State Apple Club
Ed Bardet
3700 Vineyard Ave.
Chattanooga, TN 37421

Holston Apple Computer Club
Hubert Hill
1225 Buchelew Dr.
Kingsport, TN 37663

Apple Core of Memphis
Howard Smith
2002 Adney Gap
Memphis, TN 38134

TEXAS

A.I.D.E.
P.O. Box 30878
Amarillo, TX 79120

River City Apple Corps
Tom Sethre
6108 Janey Dr.
Austin, TX 78731

K.A.O.S.
Warren Johnson
303 Hickory Ct.
Bishop, TX 78343

Dresser Atlas Apple
A. J. Dittmar
2607 Clearwood
College Station, TX 77840

Apple Corpus
Owen Hopkins, President
490 Miramar Place
Corpus Christi, TX 78411

Apple Corps of Dallas
Ed Aymond
4300 N. Central Expwy.
Dallas, TX 75206

Apple Dynamics
Glenn Weissenger
Rt. 9, Box 190, Charthouse K-2
Fort Worth, TX 76179

Gulf Coast Apple Users
Jack Turner
P.O. Box 897
Freeport, TX 77541

East Texas Computerists
Robert H. Mimlitch
10613 Old Mill Rd.
Greenville, TX 75401

H.O.T. Apple P.I.E.
Sam Redden
105 Hewitt Plaza/P.O. Box 517
Hewitt, TX 76643

HAAUG
Robin Cox
5401 Chimney Rock, #607
Houston, TX 77081

Brazosport Apple Users Group
Judy Horn
55 Pine Ct.
Lake Jackson, TX 77566

Lubbock Computer Club
Robert Darko
3002 4th St., Apt 76
Lubbock, TX 79415

San Angelo Apple Users Group
Joel McWilliams
2751 Valley Ridge
San Angelo, TX 76904

Appleseed
Larry Blodgett
6447 Stable Dr.
San Antonio, TX 78240

Temple Apple Users Group
Larry Scott
720 N. 11th St.
Temple, TX 76501

UTAH

Ogden Apple Club
Robert H. Neil, President
2567 Pringle Circle
Ogden, UT 84403

Apple Slice
O. D. Williams, President
2274 Dowington Ave.
Salt Lake City, UT 84104

Wasatch Front Apple Club
Kent Jager
5284 S. Samuel Circle
Taylorsville, UT 84118

VERMONT

Green Mountain Apple Club
Dan Sullivan, President
316 Sherwood Square
Essex Junction, VT 05452

VIRGINIA

Charlottesville Apple Core III
Universal Computers
1915 Cedar Hill Rd.
Charlottesville, VA 23666

Peninsula Apple Core
Jerry Ayacock, Secretary
200 Dover Rd.
Hampton, VA 23666

Tidewater Apple-Worms
Jerry Rowe, President
3025 Vendome Terrace
Norfolk, VA 23509

Apple T.A.R.T.
Sid Koerin
400 N. Meadow St.
Richmond, VA 23220

Novapple
Tom Lucas, Secretary
8108 Adair Lane
Springfield, VA 22151

WASHINGTON

Delta Apple Users Group
Kenneth Jordon
P.O. Box 379
APO Seattle, WA 98733

Kitsap Apple Users Group
Darrel Dunmire, President
7265 Thasos Ave. N.E.
Bremerton, WA 98310

A.P.P.L.E. Wash
Dave Lingwood, Secretary
21246 48th Ave. South
Kent, WA 98032

Port Townsend Computer
H. Mike Lane
1507 Cherry St.
Port Townsend, WA 98368

G.R.A.P.E.
3315 N.W. Parris Rd.
Poulsbo, WA 98370

Polouse Microcomputer Association
Department of Physics
Washington State University
Pullman, WA 99164

Call A.P.P.L.E (Apple Puget Sound Program Library Exchange)
304 Main Ave. S., Suite 300
Renton, WA 98055

Atomic Apple Smashers
1422 Johnston
Richland, WA 99352

Apple Users of Spokane
Vic Glassley, President
12816 E. Desmet
Spokane, WA 99216

Wenatchee Valley Apple
Mr. Neal Noble
1388 Tedford St.
Wenatchee East, WA 98801

Yakima Apple Users Group
Dale Hurd
2807 Resthaven Rd.
Yakima, WA 98901

WEST VIRGINIA

Apple Tree
Gary Anderson
P.O. Box 7396
Huntington, WV 25776

Personal Computer Club
David Rackowitz
609 56th St.
Vienna, WV 26105

WISCONSIN

Appleton Apple
Jerold Schmidt
1407 E. Harding Dr.
Appleton, WI 54915

Chi-Hi Computer Club
Warren Walicki
Colemna & Terrill Sts.
Chippewa Falls, WI 54729

RAM
Karl Spiegelberg
1821 Aspen Lane
Green Bay, WI 54303

Lax Apple Users
Charles Hosler
University of Wisconsin
1725 State St.
Lacrosse, WI 54601

Adam & Eve Apple Group
Hartley Jackson, Secretary
430 Sidney St.
Madison, WI 53703

Menomin-Apples
Alan Kelsey
802 12th Ave. E.
Menomonie, WI 54751

GOSUB 100 Computer Club
Russell Rich
Wisconsin Indianhead Technical Institute
10190 S. Knowles Ave.
New Richmond, WI 54017

Fox Valley Crab-Apples
John Oman, Mathematics Department
University of Wisconsin, Oshkosh
Oshkosh, WI 54901

WYOMING

The Apple Net
James Hassler
2203 Park Avenue Orchard Valley
Cheyenne, WY 82007

Appendix C

Vendor Names and Addresses

A B Buscaglia
2497 W. River Road
Grand Island, NY 14072

A R Davis & Co
P.O. Box 24424
San Jose, CA 95154

Aardvark Software
783 N. Water St
Milwaukee, WI 53202
Phone: 414-289-9988

Abacus Associates
6565 W. Loop S. Suite 240 Dept.11
Bellaire, TX 77401
Phone: 713-666-8146

Abacus Software
P.O. Box 7211
Grand Rapids, MI 49510
Phone: 616-241-5510

ABT Microcomputer Software
55 Wheeler Street
Cambridge, MA 02138

ABW Corp.
P.O. Box M 1047
Ann Arbor, MI 48106

Academic Hallmarks
P.O. Box 998
Durango, CO 81301

Academic Products
10476 Ilex Avenue
Pacoima, CA 91331

Accent Software
3750 Wright Place
Palo Alto, CA 94306

Access Memory
1701 West End Drive
Philadelphia, PA 19151

Access Unlimited
Dept. K3
401 N Central Exprwy
Richardson, TX 75080
Phone: 800-527-3475

ACT Educational Programs
4800 N. Shallowford Rd. S-276
Atlanta, GA 30338

Action-Research Northwest
11442 Marine View Drive S.W.
Seattle, WA 98146

Actuarial Micro Software
3915-A Valley Court
Winston-Salem, NC 27106
Phone: 919-765-5588

Addison/Wesley Publishing Co
Jacob Way
Reading, MA 01867
Phone: 617-944-3700

Addmaster Corp.
416 Junipero Serra Dr.
San Gabriel, CA 91776
Phone: 213-285-1121

ADS
7468 Maple Ave.
St. Louis, MO 63143

Advance Business Technology, Inc.
2200 S. Main St
Lombard, IL 60148
Phone: 312-629-5800

Advance Logic Systems
1195 E. Arques Ave
Sunnyvale, CA 94086
Phone: 800-538-8177

Advance Software Technology
7899 Mastin Dr.
Overland Park, KS 66204
Phone: 913-648-4442

Advanced Business Technology Inc.
12333 Saratoga-Sunnyvale
Saratoga, CA 95070

Advanced Computer Products
P.O. Box 17329
Irvine, CA 92713
Phone: 800-854-8230

Advanced Data Systems
7468 Maple Ave.
St Louis, MO 63143
Phone: 314-647-1465

Advanced Logic Systems
1195 E. Arques Ave
Sunnyvale, CA 94086
Phone: 800-538-8177

Advanced Micro Technology
60 Connolly Parkway
Hamden, CT 06514
Phone: 203-281-4359

Advanced Operating Systems
450 St. John Rd.
Michigan City, IN 46360
Phone: 219-879-4693

Advanced Software Technology Inc.
7899 Mastin Dr.
Overland Park, KS 66204

Adventure International
Box 3435
Longwood, FL 32759
Phone: 305-862-6917

Adwar Video Corp.
335W 35th St.
New York, NY 10001
Phone: 212-947-4040

Aerocomp
Redbird Airport Bldg. 8
P.O. Box 24829
Dallas, TX 75224
Phone: 800-824-7888

Afterthought Engineering
7266 Courtney Drive
San Diego, CA 92111
Phone: 714-279-2868

AG-COM
P.O. Box 706
Muscatine, IA 52761

AG*PAC
One South Park St.
Madison, WI 53715

Agricultural Mgt. & Econ. Consulting
315 Haggerty Lane
Bozeman, MT 59715

AGS Software
5217 Rock Harbor Rd
Orleans, MA 02653
Phone: 617-255-0510

Aguila Corporation
P.O. Box 330
Pepperell, MA 01463
Phone: 617-433-9502

Aids Computer Center
271 Sutter Street
San Francisco, CA 94108
Phone: 415-221-8503

Alan G. Hill Software
12092 Deerhorn Drive
Cincinnati, OH 45240

Alcor Systems
800 W. Garland Ave.
Garland, TX 75040
Phone: 214-226-4476

Aldershot's
1103 High Vista Drive
Richardson, TX 75080

Alessi Data Technology
P.O. Box 004
Needham Heights, MA 02194

Alex Massimo
4041 41 St.
San Diego, CA 92105

ALF Products
14148 Estes
Denver, CO 80215

Algol Technology Inc.
303-3 Convention Way
Redwood City, CA 94063

Algoram Computer Products Inc.
999 Independence Ave.
Mountain View, CA 94043

Allenbach Industries
4322 Manchester Ave.
Olivehain, CA 92024

Almaly Trading Corp.
Computer Products Division
7801 Coral Way Suite 116
Miami, FL 33155

Alpha-Omega Systems Inc.
1504 3rd Avenue
Seattle, WA 98101

Alphine Software
2120 Academy Circle
Colorado Springs, CO 80909

Alternative Software Inc.
1165 Barbara Drive
Cherry Hill, NJ 08003

Amdek Corp.
2201 Lively Blvd.
Elk Grove, IL 60007
Phone: 213-364-1180

American Avicultural Art & Science, Inc.
3268 Watson
St. Louis, MO 63139

American Data Cable Inc.
2864 Ray Lawyer Drive #205-352
P.O. Box 2212
Placerville, CA 95667
Phone: 916-622-3465

American Micro Products Inc.
705 N. Bowser
Richardson, TX 75081

American Mitac Corp.
3026 Scott Blvd.
Santa Clara, CA 95050

American Software & Systems
2110 N. Carson Street
Carson City, NV 89701

Amkey Inc.
220 Ballardvale St.
Wilmington, MA 01887
Phone: 617-658-7800

Ampoc Enterprises Inc.
5 Beekman St. Suite 720
New York, NY 10038

Analytic Solutions Inc.
P.O. Box 13091
Research Triangle Park, NC 27709

Analytical Engines Inc.
3415 Greystone Suite 305
Austin, TX 78731
Phone: 512-346-8430

Analytical Service Associates
21 Hollis Rd
Lynn, MA 01904
Phone: 617-593-2404

Andent
1000 North Avenue
Waukegan, IL 60085

Anderson Jacobson Inc.
521 Charcot Ave.
San Jose, CA 95131
Phone: 408-263-8520

Andrew's Spirit Inc.
P.O. Box 7811
Atlanta, GA 30357

Andromeda Computer Systems
P.O. Box 19144
Greensboro, NC 27410
Phone: 918-852-1482

Anec Software
6166 92nd Ave. S.E.
Mercer Island, WA 98040
Phone: 206-232-2694

Anthro-Digital, Inc.
103 Bartlett Avenue
Pittsfield, MA 01201
Phone: 413-448-8278

APL Press
220 California Avenue
Palo Alto, CA 94306

Apparat Inc.
4401 S. Tamarack Parkway
Denver, CO 80237

Apple Computer Inc.
20525 Mariani Ave.
Cupertino, CA 95014
Phone: 408-996-1010

Apple Country Ltd.
P.O. Box 1099
Julian, CA 92036
Phone: 619-765-0239

Apple Valley Software Inc.
13093 Everest Ave. South
Apple Valley, MN 55124

Appleware Inc.
6400 Hayes St.
Hollywood, FL 33024

Applewood Computer Products
P O Box 95271
Schaumburg, IL 60195

Applied Analytics Inc.
8910 Brookridge Dr.
Upper Marlboro, MD 20772

Applied Creative Technology, Inc.
2723 Avenue E. East
Suite 717
Arlington, TX 76011
Phone: 800-433-5373

Applied Educational Systems
RFD 2 Box 213
Dunbarton, NH 03301

Applied Engineering
P.O. Box 470301
Dallas, TX 75247
Phone: 214-492-2027

Applied Micro Systems, Inc.
P.O. Box 832
Roswell, GA 30077

Applied Software Technology
14125 Capri Drive #4
Los Gatos, CA 95030

Applitek Inc
Box 5114
Bridgeport, CT 06610
Phone: 203-378-0623

Archives
152 Boston Turnpike
Shrewsbury, MA 01545

Arizona Computer Systems
Box 125
Jerome, AZ 86331

Arken & Associates
803 Glass Drive
Leander, TX 78641

Arrow Data Systems
5910 East Washington Blvd
Los Angeles, CA 90040
Phone: 213-726-9440

ARS Publications
3710 Pacific Ave. #16
Venice, CA 90291
Phone: 213-396-9303

Artificial Intelligence Research Group
921 North La Jolla Ave.
Los Angeles, CA 90046

Artsci Inc.
5547 Satsuma Ave.
North Hollywood, CA 91601
Phone: 213-985-2922

Artworx
150 N. Main Street
Fairport, NY 14450

Ashton-Tate
3600 Wilshire Boulevard
Los Angeles, CA 90010

Assoc. For Educational Data Systems
1201 Sixteenth St. N.W.
Washington, DC 20036

Astar International
5676 Francis Ave.
Chino, CA 91710
Phone: 714-627-9887

Atlantic Software
9 Spinning Wheel Rd.
Monroe, CT 06468

Atlantis Computers
31-14 Broadway
Astoria, NY 11106
Phone: 212-728-6700

AT&T Information Systems
One Speedwell Ave.
Morristown, NJ 07960
Phone: 201-898-2000

ATS Cases
25 Washington Avenue
Natick, MA 01760

Atsuko Computing International
303 Williams Avenue
Huntsville, AL 35801

ATV Research
13A Broadway
Dakota City, NE 68731
Phone: 402-987-3771

Aurora Computer Enterprises
2423 American Ln.
Madison, WI 53704
Phone: 608-249-5875

Aurora Systems Inc.
2040 East Washington Ave.
Madison, WI 53704
Phone: 608-249-5875

Automated Simulations
1043 Kiel Ct.
Sunnyvale, CA 94088

Avalanche Productions Inc.
2460 Embarcadero Way
Palo Alto, CA 93403

Avalon Hill Game Co.
4517 Harford Road
Baltimore, MD 21214
Phone: 301-254-5300

Avant-Garde Creations
P.O.Box 30160
Eugene, OR 97403
Phone: 503-345-3043

Axiom
Barford House
Shute End Nockingham
Berkshire, U.K.

Axlon Inc.
170 N. Wolfe Rd.
Sunnyvale, CA 94086
Phone: 408-730-0216

AZD Company
Box 6471
Greenville, SC 29606
Phone: 803-297-0552

A2 Devices
P.O. Box 2226
Alameda, CA 94501

B. A. Nicholson & Co.
271 Madison Ave. Suite 1007
New York, NY 10016

Backus Data Systems Inc.
1440 Koll Circle
San Jose, CA 95112
Phone: 408-279-8711

Bizcomp Corporation
P.O. Box 7498
Menlo Park, CA 94026
Phone: 408-745-1616

B L S Inc.
2503 Fairlee Rd.
Wilmington, DE 19810
Phone: 302-478-2451

Barclay Bridge Supplies Inc.
8 Bush Avenue
Port Chester, NY 10573

Barrington Educational Computer
P.O. Box 863
Barrington, IL 60010

Basic Business Software Inc.
3300 Sirius Avenue
Suite 10
Las Vegas, NV 89102

Basic O Incorporated
P.O. Box 2029
Los Gatos, CA 95030
Phone: 408-395-4133

Bay Technical Associated Inc.
P.O. Box 387
Hwy 603
Bay St. Louis, MI 39520
Phone: 601-467-8231

BCD Association
1216 N. Black Weldor Avenue
Oklahoma City, OK 73106

BCS BL-Comm Systems
10 Yorkton Industrial Ct.
St Paul, MN 55117

Beachwood Software Division
900 Culver Road
Rochester, NY 14069

Beagle Brothers Micro Software
4315 Sierra Vista
San Diego, CA 92103
Phone: 714-296-6400

Beaman Porter Inc.
Pleasant Ridge Rd.
Harrison, NY 10528
Phone: 914-967-3504

Bell & Howell-Business Eq. Grp.
6800 N McCormick
Chicago, IL 60645

Berlinersoft
102 Jericho Turnpike
New Hyde Park, NY 11040

Bertamax Inc.
101 Nickerson Suite 550
Seattle, WA 98109

BEZ
4790 Irvine Blvd. Suite 108B
Irvine, CA 92714
Phone: 714-752-8388

Bible Research Systems
8804 Wildridge Drive
Austin, TX 78759

Big Pig Software Co, The
1548-D Adams Ave.
Costa Mesa, CA 92626

Bird Corporation
809 Terrace Mountain Drive
Austin, TX 78712

Bit 3 Computer Corp.
8120 Penn Ave. So. Suite 548
Minneapolis, MN 55431

Bitz Computer Corp.
8120 Penn Avenue South Suite 548
Minneapolis, MN 55057
Phone: 612-881-6955

Black Sun Program Exchange
P.O. Box 66
Wyncote, PA 19095
Phone: 215-884-8187

Blackbourn Inc.
10150 Crosstown Circle
Eden Prairie, MN 55344

Blue Lakes Computing
438 N. Francis Street
Madison, WI 53703

Boardroom Executive Software
255 North El Cielo Springs
Palm Springs, CA 92262

Boardwalk Products
225 South Western Ave.
Los Angeles, CA 90004

Bob Broedel
P.O. Box 20049
Tallahassee, FL 32304

Bob Sevedge
5101 N. Kostner
Chicago, IL 60630

Bob Wiseman
1899 Muskegon Dr.
Cincinnati, OH 45230

Book Company, The
11223 S. Hindry Ave. Suite 7
Los Angeles, CA 90045
Phone: 213-417-8031

Borg-Warner Educational Systems
600 West University Drive
Arlington Heights, IL 60004

Bourne & Co.
City National Center Suite 1404
Charlotte, NC 28202

BPI Solutions, Inc.
1600 W. 38th Street
Suite 44
Austin, TX 78731
Phone: 512-454-2801

BPI Systems
3423 Guadalupe
Austin, TX 78705

Brady Communications Co. Inc.
Bowie, MD 20715
Phone: 800-638-0220

Brahman Software
P.O. Box 331
Castleton, VT 05735
Phone: 802-775-2533

Brainbank Inc.
220 Fifth Ave.
New York, NY 10001

Brem Enterprises
15201 Santa Gertrudes Ave.
La Mirada, CA 90638
Phone: 714-739-5020

Brillig Systems Inc.
P.O. Box 2847
Springfield, VA 22152

Broderbund Software
1938 4th St.
San Rafael, CA 94901

Broderick & Associates
8635 Shagrock
Dallas, TX 75238

Broker Soft Inc.
P.O. Box 1762
Wilmington, NC 28402

Bruce H. Barber
11803 Rowood Drive
Houston, TX 77001

BubbleSoft
3 Mohawk Dr.
Westboro, MA 01581
Phone: 617-366-9734

Budge Company
428 Pala Avenue
Piedmont, CA 94610
Phone: 415-658-8141

Bullseye Software
Box 4845
Incline, NV 89450
Phone: 702-831-2992

Burke and Associates
1720 Los Angeles Ave. #221
Simi Valley, CA 93063
Phone: 805-584-3220

Burt Microsystems
14252 Culver Suite A259
Irvine, CA 92714
Phone: 714-559-5097

Business & Professional Software Inc.
143 Binney St.
Cambridge, MA 02142

Business Computer of Peterborough
12 School Street
Peterborough, NH 03458
Phone: 603-924-9881

Business Computer Publications Inc.
1411 St. Germain St. Suite 101
St. Cloud, MN 56301

Business Computer Services Co.
9020 Eby
Overland Park, KS 66212

Business Master
6443 Coleman Road
East Lansing, MI 48823

Byte Publications
70 Main St.
Peterborough, NH 03458

Bytes & Pieces
Box 525 Dept. J
E. Setauket, NY 11733
Phone: 516-751-2535

C & C Software
5713 Kentford Circle
Wichita, KS 67220

C & H Video
110 W. Caracas
Hershey, PA 17033

C.H. Galfo
6252 Camino Verde
San Jose, CA 95119

C.R.C. Wholesale
1439 Gordon Street
Allentown, PA 18102
Phone: 215-776-2100

Cab-Tak Inc.
Riverside St.
Nashua, NH 03062
Phone: 800-343-4311

Calendar
P.O. Box 5701
Forest Park Branch
Dayton, OH 45405

California Computer Systems
250 Caribbean Drive
Sunnyvale, CA 94086

California Pacific Computer Co.
1615 5th St. Suite B
Oakland, CA 94621

California School For the Deaf
39350 Gallaudet Drive
Fremont, CA 94538

Calsoft
346 N. Kanan Rd. #103
Agoura, CA 91301
Phone: 800-423-5290

Camp International Inc.
109 W. Washington
Jackson, MI 49204

Campbell Scientific Inc.
P.O. Box 551
Logan, UT 84321
Phone: 801-753-2342

Caner Computer Products
P.O. Box 18287
Wichita, KS 67218
Phone: 316-776-0743

Capain Software
P.O. Box 575
San Francisco, CA 94101

Capitol Computer Products
P.O. Box 416
Germantown, MD 20874
Phone: 301-972-6741

Cardio-Respiratory Management System
1300 W. Mockingbird Lane Suite 303
Dallas, TX 75247

Case Inc.
1302 Stone Ave. North
Seattle, WA 98133
Phone: 206-365-5210

Casemaker
13715 Quito Road
Saratoga, CA 95070

Cavalier Computer
1223 Camino Del Mar
Del Mar, CA 92014
Phone: 714-755-8143

Cavri Systems
26 Trumball Street
New Haven, CT 06511

C.D.S. Corp
695 East 10th North
Logan, UT 84321
Phone: 801-753-6990

CDS Cortland Data Systems
P.O. Box 14414
Chicago, IL 60614

CE Software
801 73rd St.
Des Moines, IA 50312
Phone: 515-224-1995

CECO Inc.
7654 Royce St.
Annandale, VA 22003

Centec Process Systems
Centec Bldg/11260 Roger Bacon
Reston, VA 22090

Central Point Software Inc.
P.O. Box 3563
Central Point, OR 97502

Cermetek Microelectronics
1308 Borregas Ave.
Sunnyvale, CA 94089
Phone: 408-734-8150

CESN Publications
Mountain Road
Jaffney, NH 03452

CET Research Group Ltd.
P.O. Box 2029
Norman, OK 73070
Phone: 405-360-5464

C.F. Cicciarella
502 Savannah Street
Greensboro, NC 27406

C.F. Kerchner & Associates Inc.
5507 Louise Lane East Allen
Northampton, PA 18067

Chang Laboratories
10229 North Stelling Rd.
Cupertino, CA 95014
Phone: 408-725-8088

Charles Mann & Assoc.
55722 Sant Fe Trail
Yucca Valley, CA 92284
Phone: 714-365-9718

Charles O'Neill
3 C Liberty Lane
Elk City, OK 73644

Chatsworth Data Corporation
20710 Lassen Street
Chatsworth, CA 91311

Check-Mate
51 Diauto Dr.
P.O. Box 103
Randolph, MA 02368
Phone: 800-343-7706

Checks To-Go
8384 Hercules St.
La Mesa, CA 92041

Chemron Systems Inc.
8934 L St.
Omaha, NE 68127

Cherrygarth Farms Software Inc.
101 S. Dewey St.
Auburn, IN 46706

Chestech Corporation
150 Allendale Road
King of Prussia, PA 19406

Chriswalt Publications Inc.
P.O. Box 10998 564
Austin, TX 78766

Cine-aero Productions
1821 N. Frederic St.
Burbank, CA 91505

CJM Industries
Box 436
Sterling, VA 22170
Phone: 703-435-2991

Clark Software Systems
1730 W. Mulberry Street
Shamokin, PA 17872

Clearform Company
2455 Old Middlefield Way
Mountain View, CA 94043
Phone: 415-960-1033

Clemson University
203 Freeman Hall
Clemson, SC 29631

Climate Assessment Technology Inc.
11550 Fuqua Suite 355
Houston, TX 77034

Clone Software
1446 Estes St.
Lakewood, CO 80215

CMA Micro Computer
55722 Santa Fe Trail
Yucca Valley, CA 92284

Codex
20 Cabot Blvd.
Mansfield, MA 02048
Phone: 617-364-2000

Coherent Communications Systems Corporation
60 Commerce Drive
Hauppauge, NY 11788
Phone: 516-231-1550

Collegiate Microcomputer
5500 Wabash
Terre Haute, IN 47803

Collins International Trading Corporation
23801 Calabasas Rd.
Suite 2050
Calabasas, CA 91302
Phone: 213-999-5210

Colorado Computer Peripherals
R.R. 6 Box 7-D
Golden, CO 80401
Phone: 303-278-7172

Columbia Software
5461 Marsh Hawk Way
Columbia, MD 21045

COM Press
P.O. Box 102
Wentworth, NH 03282

Combil
806 Haldemann Ave.
Joliet, IL 60436
Phone: 815-744-1095

Comdata Corporation
7900 N. Nagie Ave.
Morton Grove, IL 60053
Phone: 312-470-9600

Commercial Software Systems Inc.
7689 W. Frost Dr.
Littleton, CO 80123
Phone: 303-761-8062

Commodity Systems Inc.
200 W. Palmetto Park Road
Boca Raton, FL 33432
Phone: 800-327-0175

CommSoft
2452 Embarcadero Way
Palo Alto, CA 94303
Phone: 415-493-2184

Communitree Group
470 Castro St. Suite 207
San Francisco, CA 94114

Compak Inc.
P.O. Box 14582
Austin, TX 78766

Compco Industries Inc.
159 W. Walnut St.
Painesville, OH 44077
Phone: 216-354-4186

Comp-U-Card of America Inc.
777 Summer St.
Stanford, CT 06901

Compuclub International
P.O. Box 265
Yorba Linda, CA 92645

Compucover
P.O. Box 324
Mary Esther, FL 32569
Phone: 904-243-5793

Compukit
P.O. Box 306
Kemah, TX 77565

Compu-Law Inc.
3520 Wesley St.
Culver City, CA 90230
Phone: 213-558-3360

Compulink Corporation
1840 Industrial Circle
Longmont, CO 80501

Compumax
P.O. Box 7239
Menlo Park, CA 94025
Phone: 415-854-6700

Compu-Musik
5616 S. Quincy
Hinsdale, IL 60521
Phone: 312-323-2543

Compuserve
5000 Arlington Center Blvd.
Columbus, OH 43220
Phone: 800-545-4794
614-457-8600 in OH

Compu-Start
P.O. Box 1232
Station M
Calgary
Alberta, Canada T2P 2L2

Compu-Tations
21650 W. Eleven Mile Road
Southfield, MI 48076

Compu-Tations, Inc.
P.O. Box 502
Troy, MI 48009
Phone: 313-689-5059

Computechniques
P.O. Box 245
Fairfield, OH 45014
Phone: 513-829-5052

Computek
28278 Enderly St.
Canyon Country, CA 91351
Phone: 805-252-4244

Computer Advanced Ideas
1442 A Walnut St. Suite 341
Berkeley, CA 94709
Phone: 415-526-9100

Computer Aided & Managed Instr.
P.O. Box 2030
Goleta, CA 93118

Computer Applications Inc.
13300 SW 108 St. CLR
Miami, FL 33186
Phone: 305-385-4277

Computer Case Company
5650 Indian Mound
Columbus, OH 43213

Computer Center, The
302 Commercial
Waterloo, IA 50701
Phone: 319-232-9504

Computer Communications Specialists
6683 Jimmy Carter Blvd.
Norcross, GA 30071
Phone: 404-441-3114

Computer Components Inc.
6791 Westminister
Westminister, CA 92683

Computer Connection Ltd., The
6818 S. Yosemite St.
Englewood, CO 80122
Phone: 303-740-9360

Computer Consultants of IA Ltd.
P.O. Box 427
Marion, IA 52302
Phone: 319-373-1306

Computer Consulting
6723 E. 66th Pl.
Tulsa, OK 74133
Phone: 918-492-9036

Computer Consulting Services
1376 Cleveland St.
Clearwater, FL 33515

Computer Crossroads
3800 South Tamiami Trail
Sarasota, FL 33579

Computer Data Systems
50 N. Main St.
Logan, UT 84321
Phone: 801-753-6990

Computer Design Labs
342 Columbus Avenue
Trenton, NJ 08629

Computer Development Corporation
6700 S.W. 105th, Suite 200
Beaverton, OR 97005
Phone: 503-646-1599

Computer Distributor
P.O. Box 9194
Austin, TX 78766

Computer Emporium, The
3711 Douglas
Des Moines, IA 54312
Phone: 515-224-1992

Computer Entreprenuer Publishing Co, The
P.O. Box 456
Grand Central Station
New York, NY 10163
Phone: 800-227-3800

Computer Forum
14052 E. Firestone Blvd.
Santa Fe Springs, CA 90670

Computer Games
511 Iowa Avenue
Iowa City, IA 52240

Computer House Div/Flc Inc.
1407 Clinton Road
Jackson, MI 49202

Computer In Psychiatry/Psychology
26 Trumbull Street
New Haven, CT 06511

Computer Learning Connection
One Boston Place
Boston, MA 02108

Computer Micro Works Inc.
P.O. Box 33651
Dayton, OH 45433
Phone: 513-878-8533

Computer Outlet
Park Place-Upper Level
1095 E. Twain
Las Vegas, NV 89109
Phone: 800-634-6766

Computer Pathways Unlimited Inc.
2151 Davcor Street S.E.
Salem, OR 97302

Computer Plus
795 W. Imperial
Brea, CA 92621

Computer Products International Inc.
3225 Danny Park
Metairie, LA 70002

Computer Programs Unlimited
9710 24th Ave. SE
Everett, WA 98204
Phone: 206-337-5888

Computer Research Inc.
1879 South Main St. Suite 104
Salt Lake City, UT 84115
Phone: 801-484-7316

Computer Science Institute
9620 NE 2nd Avenue
Miami Shores, FL 33138

Computer Science Press Inc.
11 Taft Court
Rockville, MD 20850

Computer Shack
1635 S. Prairie
Pueblo, CO 81005

Computer Shop
16919A Hawthorne Blvd.
Lawndale, CA 90260
Phone: 213-371-4010

Computer Solutions of Australia
6 Maize Place
Mansfield, Queensland
Australia

Computer Sorts Systems
22713 Ventura Blvd. Suite F
Woodland Hills, CA 91364

Computer Stations Inc.
11610 Page Service Drive
St. Louis, MO 63141

Computer Systems Design
2139 Jackson Blvd.
Rapid City, SD 57701

Computer Tax Service
P.O. Box 7915
Incline Village, NV 89450

Computer Technology Associates
118 Castellano
El Paso, TX 79912
Phone: 915-581-3500

Computer Textile Inc.
10960 Wilshire Blvd.
Los Angeles, CA 90024

Computer Transceiver Systems Inc.
P.O. Box 15
East 66 Midland Ave.
Paramus, NJ 07652

Computer Translation Inc.
P.O. Box 7004 University
Provo, UT 84602

Computer Wave Inc.
6791 Westminster
Westminster, CA 92683

Computer Works Inc.
789 E. Market St.
Harrisonburg, VA 22801

Computerized Construction Mgt. Systems
27 So. Fort Lane
Layton, UT 84041

Computerized Farm Information Systems
1710 Oldridge Ave.
P.O. Box 302
Stillwater, MN 55082

Computerized Management Systems
1039 Cadiz Drive
Simi, CA 93065

Computerland
2500 B South Columbia Road
Grand Forks, ND 55082

Computermart
2984 Daytona
P.O. Box 1
Lake Havasu City, AZ 86903

Computers For Today Inc.
31368 Colinas Suite 101
Westlake Village, CA 91360

Compu-Things
270 Broadway
Revere, MA 02151
Phone: 617-289-8866

Compu-Time
P.O. Box 5343
Huntington Beach, CA 92646
Phone: 714-536-5000

Computing Concepts
P.O. Box 250
Fredonia, NY 14063

Compu-Trac
1221 Conroy St
New Orleans, LA 70115
Phone: 504-895-1474

Compu-Tron Software Service
1414 S. Fairplain Avenue
Whittier, CA 90601

Computronics Inc.
50 N. Pascack Road
Spring Valley, NY 10977
Phone: 800-431-2818

Compuvision
50 W. Brokaw Suite 64
San Jose, CA 95110
Phone: 408-947-4616

Compuware
523 S. Clewell Street
Bethlehem, PA 18015

Comshare Target Software Inc.
1935 Cliff Valley Way Suite 200
Atlanta, GA 30329
Phone: 404-634-9535

Concord Data Systems
303 Bear Hill Road
Waltham, MA 02154
Phone: 617-890-1394

Concorde Peripheral Systems, Inc.
23152 Verdugo Dr.
Laguna Hills, CA 92653
Phone: 714-859-2850

Conduit
Box 388
Iowa City, IA 52244
Phone: 319-353-5789

Connecticut Information Systems Co.
218 Huntington
Bridgeport, CT 06608

Connecticut Microcomputer
34 Del Mar Dr.
Brookfield, CT 06804
Phone: 203-775-4595

Consultant, The
414 Can-Dota
Mt Prospect, IL 60056
Phone: 312-259-7224

Context Inc.
833 West Colorado Avenue
Colorado Springs, CO 80905

Context Management Systems Inc.
23864 Hawthorne Blvd. Ste 101
Torrance, CA 90505

Continental Software
11223 S. Hindry Ave.
Los Angeles, CA 90045
Phone: 213-410-3977

Convergent Systems Inc.
245 E. 6th St. Suite 257
St Paul, MN 55101

Cook Computer Company
L905 Bailey Drive
Marshallton, IA 50158

Co-Op Software
830 4th Avenue
Geneva, IL 60134
Phone: 312-232-1984

C.O.R.P. Dynatech Microsoftware
7847 N. Caldwell
Niles, IL 60648

Corona Data Systems
31324 Via Colinas
Westlake Village, CA 91361
Phone: 213-991-1144

Corporation for Public Information, The
1714 Illinois
Lawrence, KS 66044
Phone: 913-841-3095

Corvus Systems
2029 O'Toole Ave.
San Jose, CA 95131

Counterpoint Software Inc.
Suite 140 Shelard Plaza North
Minneapolis, MN 55426
Phone: 800-328-1223

Countryside Data Inc.
718 N. Skyline
Idaho Falls, ID 83402

Cover Craft Corporation
P.O. Box 555
Amherst, NH 03031

CPAIDS
1640 Franklin Avenue
Kent, OH 44240

CPU Software
9710 24th Ave. S.E.
Everett, WA 98204
Phone: 206-337-5888

Crabapple Systems Inc.
114 Commercial St.
Portland, ME 04101

Cramapple
Box 98
Cambridge B Branch
Boston, MA 02140
Phone: 617-628-0206

Crane Software Inc.
16835 Algonquin Suite 61
Huntington Beach, CA 92649

Crawford Data Systems
P.O. Box 3000 561H
Camarillo, CA 93011
Phone: 805-484-4159

Creative Computer Peripherals Inc.
Aztec Environmental Center
1044 Lacey Road
Forked River, NJ 08731
Phone: 609-693-0002

Creative Computing
39 E. Hanover Ave.
Morris Plains, NJ 07950
Phone: 800-631-8112

Creative Discount Software
256 S. Robertson Suite 2
Beverly Hills, CA 90211

Creative Software
6081 Barbados Avenue
Cypress, CA 90630

CRIC Software Systems
15132 Stony Plain Road
Edmonton, Alberta, Canada
Phone: 403-483-5999

Criterion Systems Games
586 Kent Lane
Shoreview, MN 55112

Cross Educational Software
Box 1536
Ruston, LA 71270
Phone: 318-255-8921

Crow Ridge Associates Inc.
P.O. Box 1
New Scotland, NY 12127

Crystal Computer
17120 Monterey Road
Morgan Hill, CA 95037

Cuesta Systems Inc.
3440 Roberto Court
San Luis Obispo, CA 93401

Cuisenaire Co. of America Inc.
12 Church St. Box D
New Rochelle, NY 10805
Phone: 914-235-0900

Cyberia Inc.
2330 Lincoln Way
Ames, IA 50010

Cyber-Tech
P.O. Box 924
Conoga Park, CA 91311

Cybertronics International Inc.
999 Mount Kenable Avenue
Morristown, NJ 07960

Cyborg Corporation
55 Chapel St.
Newton, MA 02158

Dakin5 Corp
7000 Broadway Suite 304
Denver, CO 80221
Phone: 303-832-1501

Dann McCreary
Box 16435
San Diego, CA 92116

Dark Star Systems
P.O. Box 140 Dept. A
Amherst, MA 01004

Darsemeyer Electronics
450 St. John Road
Michigan City, IN 46360

Data Cast
345 Swett Road
Woodside, CA 94062

Data Compass Peripheral Products Group
2730 Regal Park Drive
Anaheim, CA 92806
Phone: 714-630-7450

Data Dynamics Technology
P.O. Box 1217
Cerritos, CA 90701

Data Frontiers Inc.
P.O. Box 92423
Rochester, NY 14692

Data Mail
P.O. Box 818
Reseda, CA 91335
Phone: 800-635-5555

Data Security Concepts
Box 31044
Des Peres, MO 63131
Phone: 314-965-5044

Data Strike
125 North 4th Street
Lacrosse, WI 54601

Data Systems Inc.
1810 Chester Avenue
Bakersfield, CA 93386

Data Transforms Inc.
616 Washington St. #106
Denver, CO 80203
Phone: 303-832-1501

Data-Basics
P.O. Box 4028
Culver City, CA 90230

Datacope
P.O. Drawer AA
Little Rock, AR 72205

Datam Consultants
P.O. Box 238
Dekalb, IL 60115
Phone: 815-758-1505

Datamost
9748 Cozycroft Ave.
Chatsworth, CA 91311
Phone: 213-366-7160

Data-Rite Corporation
3086 Wheeling Street
Aurora, CO 80011
Phone: 303-366-5267

Dataroyal Inc.
235 Main Dunstable Road
Nashua, NH 03061

Data-Soft Inc.
9421 Winnetka
Chatsworth, CA 91311
Phone: 213-701-5161

Data-Soft of NH
22 Stevens Ave.
Merrimack, NH 03054
Phone: 603-424-5217

Datassistance
40919 Paw Paw Road
Paw Paw, MI 49079

Davell Custom Software
P.O. Box 4162
Cleveland, TN 37311
Phone: 615-336-3055

David Data
12021 Wilshire Blvd. Suite 212B
Los Angeles, CA 90025
Phone: 213-478-7865

Davidson & Associates
6069 Groveoak Place #4
Rancho Palos Verdes, CA 90274
Phone: 213-378-7826

Davka Corp.
845 North Michigan Avenue
Suite 843
Chicago, IL 60611
Phone: 800-621-8227

Day & Night Computers
4738 Scotts Valley Drive
Scotts Valley, CA 95066
Phone: 408-438-5062

Decision Data Inc.
213 Lincoln Way
Ames, IA 50010

Decision Resources
44 White Birch Road
Weston, CT 06883

Decision Science Software
865 Castle Ridge Rd.
Austin, TX 78746

Decision Support Software
1438 Ironwood Drive
McLean, VA 22101

Decision Systems
P.O. Box 13006
Denton, TX 76203

Decisionmakers Inc.
3130 Impala Drive
San Jose, CA 95117

Denver Software Co.
Box 440140
Aurora, CO 80014
Phone: 303-750-9980

Desert Technology
Box 15113
Phoenix, AZ 85060

Desktech
413 I Street
P.O. Box 4
Arcata, CA 95521

Desktop Computers Software
303 Potrero St.
Santa Cruz, CA 95060

Desktop Solutions
P.O. Box 35659
Houston, TX 77235

Develcon Electronics
4037 Swamp Road
Doylestown, PA 18901
Phone: 215-348-1900

Developmental Learning Materials
One Elm Park
Allen, TX 75002

Denver Software Co., The
Box 440140
Aurora, CA 88044
Phone: 303-750-9980

Dickens Data System
478 Engle Drive
Tucker, GA 30084

Dicomed Corp.
9700 Newton Ave. South
Minneapolis, MN 55431
Phone: 612-887-7100

Digi-Scan Inc.
432 Broad Street
Shrewsburys, NJ 07701

Digital Equipment Corporation
146 Main Street
Maynard, MA 01754
Phone:617-897-5111

Digital Etc.
202 Main St.
Venice, CA 90291

Digital Images
Box 941005
Schaumburg, IL 60194
Phone: 800-222-1248

Digital Marketing
2670 Cherry Lane
Walnut Creek, CA 94596

Digital Research
801 Lighthouse
P.O. Box 579
Pacific Grove, CA 93950

Digital Word Systems
22458 Ventura Blvd.
Suite E
Woodland Hills, CA 91364
Phone: 213-992-0514

Dilithium Press
11000 S.W. 11th St.
Beaverton, OR 97005

Dip Inc.
745 Atlantic Avenue
Boston, MA 02111

Disco Tech
P.O. Box 11129
Santa Rosa, CA 95406

Disk Depot
731 W. Colorado Avenue
Colorado Springs, CO 90905

Diversified Software Research Inc.
5848 Crampton Ct.
Rockford, IL 61111
Phone: 815-877-1343

Don't Ask Computer Software
2265 Westwood Blvd.
Suite B-150
Los Angeles, CA 90064
Phone: 213-397-8811

Dormac Inc.
P.O. Box 1699
Beaverton, OR 97075
Phone: 503-641-3128

Doss Enameling Company
1224 Mariposa Street
San Francisco, CA 94107

Double Gold Software
13126 Anza Drive
Saratoga, CA 95070
Phone: 408-257-2247

Dow Jones Information Systems
P.O. Box 300
Princeton, NJ 08540

Dow Jones News/Retrieval
See Dow Jones Information Systems.

Dr. Daly's Software
Water Street
Darby, MT 59829
Phone: 406-831-2934

Dresselhaus Computer Products
Box 929
Azusa, CA 91702
Phone: 213-969-2250

Duosoft Corporation
Box 1827
Champaign, IL 61820

Dynacomp
1427 Monroe Ave.
Rochester, NY 14618
Phone: 716-442-8960

Dynamic Microprocessor Assoc.
545 Fifth Avenue Suite 602
New York, NY 10017
Phone: 212-687-7115

Dynatech Microsoftware Inc.
7847 N. Caldwell Ave.
Niles, IL 60648

Dynax
179 Riveredge Road
Tenafly, NJ 07670

Dysan Software Duplication Division
5201 Patrick Henry Drive
Santa Clara, CA 95050
Phone: 408-988-3472

Eagle Computer Furniture
6226 West Howard Street
Niles, IL 60648

Eagle Software Publishing Inc
993 Old Eage School Rd s409
Wayne, PA 19087

Earthware/Geocomp
P.O. Box 30039
Eugene, OR 97403
Phone: 503-344-0338

East Side Software Co.
344 E. 63 St. Suite 14-A
New York City, NY 10021
Phone: 212-355-2860

Eastcoast Software
110 West Caracas Ave.
Hershey, PA 17033
Phone: 717-533-8125

Eastern House Software
3239 Linda Drive
Winston-Salem, NC 27106
Phone: 919-924-2889

Eastern Software
3239 Linda Dr.
Winston-Salem, NC 27106
Phone: 919-924-2889

Eastman Pub. Co. Inc.
15720 Ventura Blvd.
Suite 222
Encino, CA 91436

Eaton Printer Products
901 South 12th Street
Watertown, WI 53094

Ecom Associates Inc.
8634 W Brown Deer Rd.
Milwaukee, WI 53223

Ecosoft
Box 68602
Indianapolis, IN 46268

Eco-Tech Inc.
2990 Lake Lansing Rd.
P.O. Box 776
East Lansing, MI 48823
Phone: 517-337-9226

Ed-Sci Development
460 Beacon St.
San Francisco, CA 94131
Phone: 415-282-7020

Edu Soft
P.O. Box 2560
Berkeley, CA 94702
Phone: 415-654-9822

Edu Tech/Innovative Educational Software
634 Commonwealth Ave.
Newton Centre, MA 02159
Phone: 617-965-4813

Edu-Ware Service, Inc.
28035 Dorothy Dr.
Agoura, CA 91303
Phone: 213-442-8960

Educational Activities Inc.
1937 Grand Ave.
Baldwin, NY 11510
Phone: 516-223-4666

Educational Computing
3144 Valentino Ct.
Oakton, VA 22124

Educational Computing Systems
136 Fairbanks
Oak Ridge, TN 37830
Phone: 615-483-4915

Educational Courseware
3 Nappa Lane
Westport, CT 06880

Educational Services Mgt. Corp.
P.O. Box 12599
Research Triangle Park, NC 27709

Educational Services of Alaska
Box 145
McGrath, AK 99627

Educational Software Professionals
38437 Grand River
Farmington Hills, MI 48018

Educational Systems Software
23720 El Torord Suite 210
El Toro, CA 92630

Educational Teaching Aids Div.
159 W. Kinzie
Chicago, IL 60610

Educational Time Sharing Systems Inc.
4122 E. Chapman Ave. Suite 28
Orange, CA 92669

Educulture
1 Dubuque Plaza Suite 150
Dubuque, IA 52001
Phone: 800-553-4858

Edutech
634 Commonwealth Avenue
Newton Center, MA 02159

Edutech Corp.
415 Cambridge #14
Palo Alto, CA 94306
Phone: 415-325-9965

Eduteck Corp.
See Edutech Corp.

Eduware Services
28035 Dorothy Drive
Agoura, CA 91301

Eiconics Inc.
200 Cruz Alta
Taos, NM 87571

Eidos Systems
2500 21st Avenue South
Nashville, TN 37212

Einstein Corporation
11340 W. Olympic Blvd.
Los Angeles, CA 90064
Phone: 213-477-4539

Eclectic Systems Corp.
P.O. Box 1166
Addison, TX 75001

Electronic Arts
2755 Campus Drive
San Mateo, CA 94403
Phone: 415-571-7171

Electronic Communications Inc.
P.O. Box 20221
Tallahassee, FL 32304

Electronic Courseware Systems
P.O. Box 2374 Station A
Champaign, IL 61820
Phone: 217-359-7099

Electronic Information Systems
669 Clay St.
San Francisco, CA 94111

Electronic Protection Devices
54 Sum Street
Waltham, MA 02154
Phone: 617-891-1813

Electronic Specialists Inc.
171 South Main Street
Box 389
Natick, MA 01760

Electronic Systems Furniture
17129 S. Dings View Avenue
Carson, CA 90746

Electrovalue Industrial Inc.
P.O. Box 157-N
Morris Plains, NJ 07950
Phone: 201-267-1117

Elliam Associates
24000 Bessemer St. Suite 47
Woodland Hills, CA 91367

Ellis Computing
3917 Noriega St.
San Francisco, CA 94122
Phone: 415-753-0186

Emerald Valley Publishing Co.
P.O. Box 5537
Eugene, OR 97405

Empire Software
387 E. Montgomery Avenue
Wynnewood, PA 19096

Encyclopedia Britannica Ed. Co.
425 N. Michigan Ave.
Chicago, IL 60611

Energy Games
649 Mission Street
San Francisco, CA 94105

Ensign Software
2313 N. Cole Road
Suite E
Boise, ID 83704
Phone: 208-378-8068

Enter Computer Inc.
6867 Nancy Ridge Drive
San Diego, CA 92121
Phone: 619-450-0601

Enterprise Systems Inc.
2333 Waukegan Road
Bannockburn, IL 60015

Epson America
3415 Kashiwa Street
Torrance, CA 90505

EPYX/Automated Simulations
1043 Kiell Ct.
Sunnyvale, CA 94086
Phone: 408-745-0700

Equipment Environments Inc.
P.O. Box 3442
87 1/2 Chandler Street
Nashua, NH 03061

ESP Computer Resources Inc.
P.O. Box 597
Hollis, NH 03049

Europress Ltd.
68 Chester Road
Hozd Grove, Stockport SK7 5NY
U.K.

Evans Newton Inc.
7335 E. Acoma Drive
Scottsdale, AZ 85260

Eventide Clockworks Inc.
265 West 54th Street
New York, NY 10001

Evolution Software Inc.
1632 Bathurst St
Toronto, Ontario, Canada M5P3J5

Exec Systems
P.O. Box 192
Clinton, MD 20735
Phone: 301-868-5487

Executive Data Systems Inc.
290 Interstate North
Atlanta, GA 30339

Executive Peripheral Systems Inc.
800 San Antonio Rd.
Palo Alto, CA 94303

Execuware
4530 Park Rd. Suite 348
Charlotte, NC 28209
Phone: 704-525-9881

Expotek
2723 W. Windrose Suite 3
Phoenix, AZ 85029

Express Marketing
Box 1736
Poulsbo, WA 98370

Facts Management Systems Inc.
302 Bird Road
Coral Gables, FL 33146
Phone: 800-327-7210

Far West Laboratory For Educ. Research
1855 Falson Street
San Francisco, CA 94103

Farmplan Computer Systems Inc.
1055 Sunnyvale-Saratoga
Sunnyvale, CA 92626

Fast Louie's Discount Apple Software
P.O. Box 1495
Auburn, CA 95603

Fast-Track Computer Products
Box 456
Columbus, OH 43216

FCI Inc.
1407 Clinton Rd.
Jackson, MI 49202
Phone: 517-782-2132

Fiberbilt Ikelheimer-Ernst Inc.
8601 W. 26th St.
New York, NY 10001
Phone: 212-675-5820

Financial Software Inc.
11401 Westridge Circle
Chardon, OH 44024

Financial Systems Inc.
P.O. Box 2021
Kearney, NE 68847

Firefly Computing
2075 N. Stadium Lane
Provo, UT 84601

First Software Inc.
5622 East Presidio
Scottsdale, AZ 85200

Flight Opps
3587 Birch St. #237
Newport Beach, CA 92660

FlipTrack Learning Systems
526 N. Main St. Dept. IC-B
P.O. Box 711
Glen Ellyn, IL 60137
Phone: 312-790-1117

FlipTrack Training Tapes
526 N. Main Street
Box 71
Glen Ellyn, IL 60137

FM Panatronics Corp.
Box 1088
Stone Mountain, GA 30086

FMJ Inc.
P.O. Box 5281
Torrance, CA 90510

David Foard
P.O. Box 5701
Dayton, OH 45405

Foggle Computing Corp.
P.O. Box 5166
Spartanburg, SC 29304

Football
80 N. High St
Chillicothe, OH 45601

Frank Salvage
5101 N. Kostner St.
Chicago, IL 60360

Franklin Computer Corporation
2138 Route 38
Cherry Hill, NJ 08002
Phone: 609-482-5900

Franklin Software
1900 South 25th Street
Lincoln, NE 68502

Frederick E. Deeg and Associates
13234-A Fiji Way
Marina Del Rey, CA 90291
Phone: 213-823-0767

Friendly Software
302 Tobett Suite 138
Richland, WA 99352

Fullmer Associates
1132 Via Jose
San Jose, CA 95120

Funk Vocab-Ware
4825 Province Line Road
Princeton, NJ 08540

Furn Wood Manufacturing Inc.
5664 S.W. Carmen Drive
Lake Oswego, OR 97034

Future Solutions Inc.
3198-H Airport Loop Dr.
Costa Mesa, CA 92626

Future Tech Distributors
P.O. Box 50878
Palo Alto, CA 94303

GAF Software
127 Mount Spring Road
Tolland, CT 06084

Galaxy
P.O. Box 22072
San Diego, CA 92122

Gandalf
1019 S. Noel Ave.
Wheeling, Il 60090
Phone: 312-541-6060

Garbo Software
212 W. Wood #25
Carlsbad, NM 88220

Gebelli Software
1771 Tribute Rd. Suite A
Sacramento, CA 95815
Phone: 916-925-1432

Geegery Software Works, The
Box 8028
Des Moines, IA 50301

General Technical Corporation
4101 N. St. Joseph Avenue
Evansville, IN 47712

General Videotex Corp.
377 Putnam Avenue
Cambridge, MA 02139

George Earl
1302 S.General MacMullen
San Antonio, TX 78237
Phone: 512-434-3681

George Otis Companies
7710 Computer Avenue
Edina, MN 55435

George W. Hallahm & Co. Inc.
1600 Soldiers Field Rd.
Boston, MA 02135
Phone: 617-782-6100

Geotec
1920 NW Milford Way
Seattle, WA 98177

Gesu Ent
15161 Vermont St.
Westminster, CA 92683

Gibson Laboratories
23192-D Verdugo Dr.
Laguna Hills, CA 92653
Phone: 714-770-3088

Golden Empire Publications Inc.
1919 E. Sycamore
Anaheim, CA 92805

Golden Knight Software
11 Lark Lane N.
Huntington, CT 06484
Phone: 203-929-6004

Golway Computer Enterprises
350 Richard Avenue
Staten Island, NY 10309

Gordon Publications Inc.
P.O. Box CD
Winter Beach, FL 32971

Gourmet Software
671 Eden Avenue
San Jose, CA 95117

Great Divide Software Inc.
8060 West Woodard Drive
Lakewood, CO 80227

Great Lakes Digital Resources
P.O. Box 32133
Detroit, MI 48232

Great Plains Software
123 North 15th Street
Fargo, ND 58102

Group Technology Ltd.
Check, VA 24072
Phone: 703-651-3153

Grover & Associates
7 Mt. Lassen Drive Suite D11
San Rafael, CA 94903

Gryphon Microproducts
P.O. Box 6543
Silver Spring, MD 20906
Phone: 301-946-2585

Gryphon Systems
P.O. Box 755
Cardiff, CA 92007

GS Computer Enterprises
P.O. Box 8050
Ann Arbor, MI 48107
Phone: 313-665-6416

Guardian Software
P.O. Box 2
Highwood, IL 60040

Gusdorf Corp
6900 Manchester Ave.
St Louis, MO 63143

H & H Scientific
13507 Pendleton St.
Ft Washington, MD 20744
Phone: 301-292-3100

H & H Trading Co.
P.O. Box 549
Clayton, CA 94517

H & W Computer Systems
P.O. Box 1207
Pahrump, NV 89041
Phone: 703-727-4274

H.A.L. Labs
7074 Midland Rd. Suite 23
Riverside, CA 92505
Phone: 714-359-8480

Happ Electronics Inc.
4640 Island View
Oshkosh, WI 5490

Hardcore Computing
Dept. 7 P.O. Box 44549
Tacoma, WA 98444

Harris Laboratories Inc.
624 Peach Street
P.O. Box 80837
Lincoln, NE 68501

Harrisonburg Computer Services
444 Meyers Avenue
Harrisonburg, VA 22801

Hartley Courseware Inc.
P.O. Box 431
Dimondale, MI 48821

Harvest Computer Systems
203 West 11th Street
Alexandria, IN 46001

Hayden
600 Suffolk
Low Orwell, MA 08153
Phone: 617-937-0200

Hayden Book Company
50 Essex Street
Rochelle Park, NJ 07662
Phone: 800-631-0856

Hayes Computer Systems
313 North Main Street
Monroe, NC 28110

Hayes Microcomputer Products Inc.
5835 Peachtree Corners East
Norcross, GA 30092
Phone: 404-449-8791

H.C.I.
1115 W. Hwy 7
Hutchinson, MN 55350

Hearst Corp, The
Box 10076
Des Moines, IA 50350

Hewlett-Packard
16399 Bernardo Drive
San Diego, CA 92127

High Order MicroElectronics Corp.
17 River St.
Chagrin Falls, OH 44022
Phone: 216-247-3110

High Technology Inc.
Box 14665
Oklahoma City, OK 73113
Phone: 405-478-2105

High Technology Software Prod.
P.O. Box 60406
Oklahoma City, OK 73106
Phone: 405-478-2105

Highland Computer Services
6713 Vista Del Mar
La Jolla, CA 92037
Phone: 714-454-5079

Highlands Computer Services
14422 S.E. 132nd
Renton, WA 98056
Phone: 206-228-6691

Hi-Plains Systems Inc.
P.O. Box 8152
Amarillo, TX 79109

Hoffman Educational Systems
1720 Flower Avenue
Duarte, CA 91010

Home Computer Software
1307 S. Mary Ave. #209
Sunnyvale, CA 94087
Phone: 408-735-8400

Homeware
P.O. Box 3062
Ruston, LA 71272

Horizon Simulations
7561 Crater Lake Hwy
White City, OR 97503
Phone: 503-826-4640

Houston Instrument
P.O. Box 15720
Austin, TX 78761

Howard Software Services
8008 Girard Avenue Suite 310
La Jolla, CA 92037

Howard W. Sams & Co. Inc.
4300 West 62nd Street
P.O. Box 7092
Indianapolis, IN 46206
Phone: 317-298-5400

Human Systems Dynamics
9010 Reseda Blvd.
Suite 222
Northridge, CA 91324
Phone: 213-993-8536

Huntington Computing
P.O. Box 1297
Corcoran, CA 93212
Phone: 800-692-4146

I O T C Inc.
910 Sully
Laramie, WY 82070
Phone: 307-721-5818

ICON Computer Corp.
81 Plymouth Road
Plainview, NY 11803
Phone: 516-433-6745

Ideal Systems
P.O. Box 681
Fairfield, IA 52556
Phone: 515-472-5979

Ideatech Co.
P.O. Box 62451
Sunnyvale, CA 94088

IEM Screening Systems Inc.
10440 Burbank Blvd.
North Hollywood, CA 91601
Phone: 213-877-2182

Ikelheimer-Ernst Inc.
601 West 23rd Street
New York, NY 10001

Image Computer Products, Inc.
615 Academy Dr.
Northbrook, IL 60062
Phone: 312-564-5060

Incomm
115 N. Wolf Road
Wheeling, Il 60090
Phone: 312-459-8881

Independence Software
1324 Second Ave.
New York, NY 10021

Indian Head Software
1002 Indian Head Drive
Snow Hill, NC 28580

Individualized Operand
P.O. Box 3030
San Rafael, CA 94912

Infocom Inc.
55 Wheeler St
Cambridge, MA 02138
Phone: 617-492-1031

Information Synergy Inc.
P.O. Box 992
Acton, MA 01720

Information Unlimited Software Inc.
281 Arlington Ave.
Berkeley, CA 94707
Phone: 415-525-9452

INMAC
Dept 12-2465 Augustine Dr.
Santa Clara, CA 95051
Phone: 408-727-1970

Innovative Data Technology
4060 Morena Blvd.
San Diego, CA 92117
Phone: 619-270-3990

Innovative Design Software Inc.
920 1st National Bank Tower
Las Cruces, NM 88001
Phone: 505-522-7373

Innovative Program Associates
One Airport Place
Princeton, NJ 08540
Phone: 609-924-7272

Innovative Software
260 Sheridan Drive
Palo Alto, CA 94306

Insoft, Inc.
10175 S.W. Barbur Blvd. Suite 201-B
Portland, OR 97219
Phone: 503-244-4181

Inst Sys/School Dist Of Philadelphia
5th & Lucerne Street
Philadelphia, PA 19140

Instant Software
Elm Street
Peterborough, NH 03458
Phone: 603-924-9471

Institute for Computers Jewish Life
845 N. Michigan Ave. Suite 843
Chicago, IL 60611

Institute for Econometric Research, The
3471 N.Federal Highway
Ft. Lauderdale, FL 33306
Phone: 800-327-6720

Institute Of Industrial Engineers
25 Technology Park/Atlanta
Norcross, GA 30092

Instructional/Comm Tech Inc.
10 Stepar Place
Huntington, NY 11746

Insurance Micro Software Inc.
P.O. Box 13318
Wichita, KS 67213
Phone: 316-265-5695

Integral Data Systems Inc.
Route 13
Milford, NH 03055

Integrated Design Engineering
645 Axminister Drive
Fenton, MO 63026
Phone: 314-343-0005

Intelligent Computer System
722 S. 24th St.
Arlington, VA 22202
Phone: 703-684-0738

Intelligent Investor, The
270 N. Canon Dr.
Beverly Hills, CA 90210
Phone: 213-938-5461

Intentional Educations Inc.
51 Spring Street
Watertown, MA 02172

Interactive Fiction
P.O. Box 603
Inverness, CA 94937

Interactive Microware Inc.
P.O. Box 771
State College, PA 16801

Interactive Sports Systems
1022 Harmony Street
New Orleans, LA 70175

Interactive Structures Inc.
146 Montgomery Avenue
Bala Cynwyd, PA 19004

Interlude
10635 Richmond
Houston, TX 77042

International Apple Core Inc.
910 A. George Street
Santa Clara, CA 95050

International Software Enterprise
350 W. Sagamere Pkwy
West LaFayette, IN 47906

International Software Marketing
120 E. Washington St. Suite 421
Syracuse, NY 13202

Interpretive Education
2306 Winters Drive
Kalamazoo, MI 49002

Intra Computer
101 W. 31st St.
New York, NY 10001
Phone: 212-947-5533

Intro Computer Inc
101 W. 31st St.
New York, NY 10001

Investment Evaluation Software
9192 W. Belmont Ave.
Littleton, CO 80123
Phone: 303-978-0261

Investors Micro Messenger
Box 319
Harvard, MA 01451
Phone: 617-456-8830

Investors Software
Box 2605
San Francisco, CA 94126
Phone: 415-981-5261

Isaac Software
841 West Nebraska Ave.
St.Paul, MN 55117

Island Graphics, Inc.
P.O. Box V
Bethel Island, CA 94511
Phone: 415-684-2664

ITC Technologies Corp.
7100 Boulevard East 21
Guttenberg, NJ 07093

Ivers Specialities
Dept. IC-1 Box 524
12841 Hawthorne Blvd.
Hawthorne, CA 90250

J & S Software
140 Reid Avenue
Port Washington, NY 11050

Jade Computer Products
4901 W. Rosecrans
Hawthorne, CA 90250
Phone: 213-973-7707

Jagdstaffel Software
618 Blossom Hill Road
San Jose, CA 95123

Jamestown Publishers
P.O. Box 6743
Providence, RI 02940

Jay Rosenberg
5 Fraser Ave.
Monticello, NY 12701

Jaylet Co.
P.O. Box 607
Bloomfield, CT 06002

John Bell Engineering Inc.
1014 Center St.
San Carlos, CA 94070
Phone: 415-592-8411

John Wiley & Sons Inc.
605 Third Avenue
New York, NY 10158

Johnny Appleseed Software and Company
P.O. Box 7000-287
Palos Verdes Pennisula, CA 90274

K & R Data Works
P.O. Box 228
Aurora, IL 60507

Kane Computing
184 Pine Brook Blvd.
New Rochelle, NY 10804

Keith Harrel
1609 Hedgefield Ct.
Tallahassee, FL 32303

Kennedietsch Productions
502 Jackson
Oak Park, Il 60304
Phone: 312-848-7352

Kensington Microwave Ltd.
300 E. 54th St.
Suite 3L
New York, NY 10022
Phone: 212-486-2802

Kern Publications
190 Duck Hill Road
Duxbury, MA 02332

Key Systems Inc.
16 Ocean East
Marathon, FL 33050

Key Tronic
Dept. E1
P.O. Box 14687
Spokane, WA 99214

Keyboard Company, The
7151 Patterson Dr.
Garden Grove, CA 92641
Phone: 714-891-5831

Killer T-Cell
Mdah Box 6
6723 Bertner Ave.
Houston, TX 77030

Kiloware
P.O. Box 5265
Valparaiso, IN 46405

Kleinhammer Business Software
P.O. Box 1065
Morro Bay, CA 93442

Knickerbocker Case Company
2300 West Wabansia
Chicago, IL 60647

Knossos Inc.
422 Redwood Ave.
Corte Madera, CA 94925

Korsmeyer Electronic Design
16411 Del Mar Lane
Huntington Beach, CA 92649

Kraft Systems Co.
Box 1268
Vista, CA 92083
Phone: 714-724-7146

Krell Software
1320 Stonybrook Rd.
Stonybrook, NY 11790
Phone: 516-751-5139

L & S Computerware
1589 Fraser Dr.
Sunnyvale, CA 94087
Phone: 408-738-3416

Lahart
736 Weadley Road
King Of Prussia, PA 19406

Last
619 Sarver Avenue
Montgomery, AL 36109

Last Electronics
P.O. Box 1300
San Andreas, CA 95249

Laumer Research
1832 School Road
Carrollton, TX 75006

Laurel Software
1310 College Avenue
Boulder, CO 80302

Lazer Systems
1791 Captial
Corona, CA 91720

Leading Edge Products Inc.
225 Turnpike Street
Canton, MA 02021

Learning Company, The
4370 Alpine Rd.
Portola Valley, CA 94025
Phone: 415-851-3160

Learning Tools
686 Massachusetts Avenue
Cambridge, MA 02139
Phone: 617-566-7585

Learning Well
200 South Service Rd.
Roslyn Heights, NY 11577

Legend Industries Ltd.
2220 Scott Lake Rd.
Pontiac, MI 48054
Phone: 313-674-0953

Lehigh Valley Computer Corp.
523 South Clewell St.
Bethlehem, PA 18015

LeMar Instruments
2107 Artesia Blvd.
Redondo Beach, CA 90278
Phone: 213-374-1673

Lewis Video Productions
601 West End Avenue
New York, NY 10024
Phone: 212-496-0223

Lexicon Corporation
1541 N.W. 65th Ave.
Ft. Lauderdale, FL 33313
Phone: 305-792-4400

Life Science Associates
1 Fenimore Rd. P.O. Box 500
Bayport, NY 11705
Phone: 516-472-2111

Lifeboat Associates
1651 Third Avenue
New York, NY 10028

Lighting Software
Box 11725
Palo Alto, CA 94306
Phone: 415-327-3280

Link Systems
1640 19th St.
Santa Monica, CA 90404
Phone: 213-453-1851

Little Brown and Company
34 Beacon St.
Boston, MA 02106

Little Genius Ltd.
324 Regent Street
Albany House 504
London, England

LJK Inc.
7852 Big Bend Blvd.
St. Louis, MO 63119
Phone: 314-962-1855

Loadstar Technology Corp.
10073 Saich Way
Cupertino, CA 95014

Locus Systems
P.O. Box 248
North Wilkesboro, NC 28659
Phone: 919-838-4166

Logical Choice, The
9200 Baltimore National Pike
Ellicott City, MD 21043
Phone: 301-465-3175

Logical Software
P.O. Box 354
Farmington, MI 48024

Lotus
180 Franklin Street
Cambridge, MA 02139

Louis K Bell
P.O. Box 7
Augusta, GA 30903

Ludlow Composites
2100 Commerce Drive/Indstrl Pk
Fremont, OH 43420
Phone: 419-332-5531

Luster Software Services
8401 Fountain Ave. Suite 16 Box N
Los Angeles, CA 90069

M & M Software Inc.
380 N. Armando #Z-19
Anaheim, CA 92806

M & R Engineering Company
4730 W. Addison
Chicago, IL 60641

M & R Enterprises
910 George St.
Santa Clara, CA 95050
Phone: 408-980-0160

Mac Software
P.O. Box 27
Chillicothe, OH 45601

Macromaty
P.O. Box 4729
Margate, FL 33063

Macrotronics Inc.
1125 Golden State Blvd.
Turlock, CA 95380

MadWest Software
121 N. Allen Street
Madison, WI 53705
Phone: 608-238-4875

Magna Soft
5068 Princess Ann Rd.
La Canada, CA 91011
Phone: 213-790-9817

Mail Order Software
P.O. Box 5004
Toledo, OH 43611

Malibu Microcomputing
23910-A Deville Way J
Malibu, CA 90265

Mannee Medical Inc.
3483 Goldengate Way Suite 11
Lafayette, CA 94549

Mannesmann Tally
8301 S. 180th St.
Kent, WA 98032

Mark of the Unicorn
P.O. Box 423
Arlington, MA 02174
Phone: 617-489-1387

Mark/Donn Associates
Box 1589
Marietta, GA 30061
Phone: 404-422-8169

Maromaty
P.O. Box 4729
Margate, FL 33063

Marshall Associates
Mastersort/Dept 102
P.O. Box 12042
Huntsville, AL 35803
Phone: 205-881-7578

Mast Development Co.
2212 East 12th St.
Davenport, IA 52803

Mastertype
P.O. Box 5223
Stanford, CA 94305

Masterworks Software
15 Dapplegray Ln.
Rolling Hills, CA 90274
Phone: 213-530-3063

Math Software
1233 Blackthorn Place
Deerfield, IL 60015

Mathware/Math City
4040 Palos Verdes Dr. North
Rolling Hills, CA 90274
Phone: 213-541-3377

Matrix Software
315 Marion
Big Rapids, MI 49307

MCE Inc.
157 S Kalamazoo Mall
Kalamazoo, MI 49007

McGraw-Hill Inc.
P.O. Box 397
Hancock, NH 03449

McGraw-Hill Book Co.
P.O. Box 400
Hightstown, NJ 08520

McGraw-Hill Book Co. Coll Div.
1221 Avenue of the Americas
New York, NY 10020

McMullen & McMullen Inc.
P.O. Box 230 Perry St.
Jefferson Valley, NY 10535
Phone: 914-245-2734

McPheters Wolf & Jones
16704 Marguardt Avenue
Cerritos, CA 90701

MCS Micro Control Systems
143 Tunnel Rd.
Vernon, CT 06066

MD Processor II
7225 Clayton Road
St. Louis, MO 63117

M.D. Software
7225 Clayton Road
St. Louis, MO 63117

MDC Micro Data Collection Inc
P.O. Box 115
Novato, CA 94948
Phone: 415-666-1322

Mean
256 N. Washington St.
Falls Church, VA 22046

Mechanix Illustrated
1515 Broadway
New York, NY 10036
Phone: 212-719-6572

MED Data
180 Allen Road N.E. Suite 210 N.
Atlanta, GA 30328

MED Systems Software
Box 3558
Chapel Hill, NC 27514
Phone: 919-933-1990

Media Service Concepts
1713 N. North Park Avenue
Chicago, IL 60614

Medi-Byte
4699 S.W. 45th Street Dept IC
Fort Lauderdale, FL 33314
Phone: 305-587-1130

Medical Office Systems Inc.
205 N. Frederick Avenue
Gaithersburg, MD 20760

Medical Watch Software
1620 Ensenada Dr.
Modesta, CA 95359

Medi-Sim Inc.
350 Union Station
Kansas City, MO 64108

Mehrtens Financial Data Systems
P.O. Box 45
Ardsley, NY 10502
Phone: 914-693-8546

Memory Enterprises
1411 Kalmia Ave.
Boulder, CO 80302
Phone: 303-449-6454

Memory Technologies
6400 Barrie Rd.
Edina, MN 55435

Mentor Software Inc.
Box 791
Anoka, MN 55303

Merrimack Systems
Box 5218
Redwood City, CA 94003

Merry Bee Communications
815 Crest Drive
Omaha, NE 68046

Mesa Research
Rt. 1-1456A
Waco, TX 76710
Phone: 800-824-7888

Metacomet Software
P.O. Box 31337
Hartford, CT 06103

Metamorphic Systems
Box 1541
Boulder, CO 80306
Phone: 303-499-6502

MFJ Enterprises
P.O. Box 494
Mississippi State, MS 39762
Phone: 800-647-1800

Micom Systems Inc.
Silicon Valley Development Center
3393 Dela Cruz Blvd.
Santa Clara, CA 95050
Phone: 408-727-5275

Micro Age
R.R. 1
Princeton, IA 52766

Micro Associates
2300 Highway
365 Suite 5
Nederland, TX 77627

Micro Austin
3187 Redstone Ln.
Boulder, CO 80303

Micro D
17406 Mt. Cliffwood Circle
Fountain Valley, CA 92708
Phone: 714-540-4781

Micro Data Systems Inc.
Market Square Box 248
Lafayette, IN 47902
Phone: 317-448-1616

Micro Decision Systems
130 Fox Crost Rd.
Pittsburgh, PA 15220
Phone: 412-276-2387

Micro Display Systems Inc.
P.O. Box 455
Hastings, MN 55033
Phone: 612-437-2233

Micro Focus
1602 Civic Center Drive
Santa Clara, CA 95050

Micro Ink Inc.
P.O. Box 6502
Chelmford, MA 01824

Micro Lab
2310 Skokie Valley Rd.
Highland Park, IL 60035
Phone: 312-433-7550

Micro Learningware
P.O. Box 2134
North Mankato, MN 56001

Micro Lipps
2210 Wilshire Blvd. Suite 705
Santa Monica, CA 90403
Phone: 213-459-3177

Micro Logic Corp.
Box 174
Hackensack, NJ 07602
Phone: 201-342-6518

Micro Management Systems
2803 Thomasville Rd. E
Cairo, GA 31728

Micro Motion
12077 Wilshire Blvd. Suite 506
Los Angeles, CA 90025
Phone: 213-821-4340

Micro Mountain
14617 NE 169th Street
Woodinville, WA 98072
Phone: 206-483-2000

Micro Music Inc.
P.O. Box 386
Normal, IL 61761

Micro Mystique
73-20 183rd Street
Flushing, NY 11366
Phone: 212-454-6190

Micro Peripherals Inc.
4426 S. Century Drive
Salt Lake City, UT 84107

Micro Power & Light Company
12820 Hillcrest Rd. Suite 224
Dallas, TX 75230
Phone: 214-239-6620

Micro Program Designs
5440 Crestline Road
Wilmington, DE 19808

Micro Software Systems
7927 Jones Branch Drive
McLean, VA 22102

Micro Tsp
928 Mears Court
Stanford, CA 94305

Microcom Inc.
1400-A Providence Hwy
Norwood, MA 02062
Phone: 617-762-9310

Micro-Comp Inc.
2015 NW Circle Blvd.
Corvallis, OR 97330

Microcomputer Technology Inc.
3304 W. MacArthur Blvd.
Santa Ana, CA 92701

Microcomputing Research
29 Estancial Drive
Marana, AZ 85238

Microcontrol Systems
143 Tunnel Road
Vernon, CT 06066

Microcraft Systems Inc.
188 Kerby Road
Grosse Point Farms, MI 48236

Micro-Design
6301 Manchaca Road Suite J
Austin, TX 78745

Micro-Ed
31 Marshall Dr.
Edison, NJ 08817
Phone: 201-572-3415

Micro-Grip
3164 Dumbarton Ave.
San Bernardino, CA 92404
Phone: 714-864-6643

Micromate Electronics Inc.
2094 Front St.
East Meadow, NY 11554
Phone: 516-794-1072

Micromatics Inc.
181 No. 200 W. Suite 5
Bountiful, UT 84010

Micromation Limited
1 Yorkdale Rd. Suite 406
Toronto, Ontario, Canada M6A 3A1
Phone: 416-781-6675

Micromax Systems
6868 Nancy Ridge Drive
San Diego, CA 92121
Phone: 619-457-3131

Micromotion
12077 Wilshire Blvd.
Los Angeles, CA 90025

Micromp Civil Engineering
P.O. Box 630
Solana Beach, CA 92075

Microphys Programs
2048 Ford Street
Brooklyn, NY 11229

Micropi
2445 North Nugent
Lummi Island, WA 98262

MicroPro International
33 San Pablo Ave.
San Rafael, CA 94903
Phone: 415-457-8990

Microproducts
24627 Watt Road
Ramona, CA 92065
Phone: 619-789-6510

Micro-Sci
2158 South Hathaway Street
Santa Ana, CA 92705
Phone: 714-662-2801

Micro-Security Systems
P.O. Box 457
Blairsville, PA 15717
Phone: 412-459-7671

Microsoft Consumer Products
400 108th Ave. N.E.
Bellevue, WA 98004
Phone: 206-454-1315

Microsource
1425 W. 12th Place
Tempe, AZ 85281

Micro-Sparc Inc.
P.O. Box 325
Lincoln, MA 01773

Microstand
Box 96
Tolovana Park, OR 97145
Phone: 503-436-1429

Microstat Software
Box 172
Concord, MA 01742

Microstuf Inc.
Box 3337
Decatur, GA 30033
Phone: 404-491-3787

Micro/Sys
1367 Foothill Blvd.
La Canada, CA 91011

Microtek
9514 Chesapeake Dr.
San Diego, CA 92123
Phone: 714-278-0633

Microware Associates Inc.
220 East 50th Street
New York, NY 10022
Phone: 212-308-9593

Micro-Ware Dist, Inc.
1342 B Route 23
Butler, NJ 07405
Phone: 201-839-3487

Midwest Data Source Inc.
1010 Nimitz Rd.
Cincinnati, OH 45230
Phone: 513-231-2023

Midwest Software Associates
P.O. Box 301
St. Ann, MO 63074

Mighty Byte Computer Inc.
12629 Tatura Blvd Suite 555
Phoenix, AZ 85032

Mike Iverson
#1 Jo-Lin Ct
El Sobrante, CA 94809

Mike Piaser Company, The
13400 Thraves Road
Garfield Heights, OH 44125

Mike Rowe Productions
Box 43504
Tucson, AZ 85733

Milliken Publishing
1100 Research Blvd.
St. Louis, MO 63132
Phone: 314-991-4220

Milton Bradley Educational Software
Shaker Road
East Longmeadow, MA 01028
Phone: 413-525-6411

Mimco Inc.
1547 Cunard Rd.
Columbus, OH 43227
Phone: 614-237-3380

Mimic Inc.
Box 921
Acton, MA 01720
Phone: 617-263-2101

Mind Systems Corp.
Box 506
Northampton, MA 01061
Phone: 413-586-6463

Min Microcomputer Software
5835-A Peachtree Corners
Norcross, GA 30092
Phone: 404-447-4322

Minit Man Printing
134B West Allegan St.
Otsego, MI 49078

Minnesota Educ. Computing Consortium
2520 Broadway Dr.
St. Paul, MN 55113
Phone: 612-376-1118

Mint Software
6422 Peggy Drive
Baton Rouge, LA 70806
Phone: 504-766-2318

MJK Associates
122 Saratoga Ave. Suite 11
Santa Clara, CA 95050
Phone: 408-247-5102

Moneydisk
Box 1531
Richland, WA 99352
Phone: 509-943-0198

Moneybee
6900 East Camelback Rd.
Scottsdale, AZ 85251

Monument Computer Service
Box 603
Joshua Tree, CA 92284
Phone: 800-854-0561

Morgan-Fairfield Graphics
P.O. Box 5457
Seattle, WA 98105
Phone: 206-632-1374

Morningstar
2535 College Avenue #301
Berkeley, CA 94704
Phone: 415-845-9111

Mountain Computer Inc.
300 El Pueblo Rd.
Scotts Valley, CA 95066
Phone: 408-438-6650

Mountain Hardware
300 Harver West Blvd.
Santa Cruz, CA 95060
Phone: 408-429-8600

MPA Enterprises
Box 6020
Wyomissing, PA 19610

MSSS D Inc.
3412 Binkley
Dallas, TX 75205
Phone: 214-522-8051

Multi-Media Video Inc.
3350 Scott Blvd. Bldg. 21
Santa Clara, CA 95051
Phone: 408-727-1733

Multi-tech Systems Inc.
82-Second Avenue SE
New Brighton, MN 55112
Phone: 612-631-3550

Muse Software
347 N. Charles St.
Baltimore, MD 21201
Phone: 301-659-7212

Mytopia Gameware Institute
Box 625
Sioux City, IA 51102

Narom Industries
1264 Deer Trail Line
Libertyville, IL 60048

Nat Hellman III, Inc.
5951 E. Firestone Blvd.
South Gate, CA 90280
Phone: 213-773-3576

National Business Institute
407 Galloway St.
Eau Claire, WI 54701

National Educational Software
1879 Locust Drive
Verona, WI 53593
Phone: 608-845-8410

National Home Computer Game Exchange
P.O. Box 20929
Columbus, OH 43220

Natural Language Systems
411 Barber Ave.
Ann Arbor, MI 48103

Navic Software
Box 14727
North Palm Beach, FL 33408
Phone: 800-327-2133

NC S & H Software
Box 5
Manvel, NV 58256
Phone: 201-768-9274

Nebs Computer Forms
78 Hollis Street
Groton, MA 01450
Phone: 617-448-6167

NEC Information Systems
5 Militia Drive
Lexington, MA 02173

Nestar Systems, Inc.
2585 East Bayshore Rd.
Palo Alto, CA 94308
Phone: 415-493-2223

Netronics R & D Ltd.
333 Litchfield Road
New Milford, CT 06776
Phone: 800-243-7428

Nibble
Box 325
Lincoln, MA 01773
Phone: 617-259-9710

Nibble Notch
4211 N.W. 75th Terrace
Lauderhill, FL 33319

Nibble/Micro-Sparc Ind
P.O. Box 325
Lincoln, MA 01773
Phone: 617-259-9710

Nikrom Technical Products
25 Prospect Street
Leominster, MA 01453
Phone: 617-537-9970

Nilonel Mfg.
5 Stevens Rd.
Worcester, MA 01603

Nonagon Software
P.O. Box 1486
El Cerrito, CA 94530
Phone: 415-237-4406

Norell Data Systems
3400 Wilshire Blvd. Box 70127
Los Angeles, CA 90010

Northern Technology Books
P.O. Box 62
Evenston, IL 60204

Notable Software
P.O. Box 1556
Philadelphia, PA 19105

Novation Inc.
18664 Oxnard Street
Tarzana, CA 91356
Phone: 213-996-5000

NTS Software
680 Arrowhead Ave.
Rialto, CA 92376

Number Nine Inc.
Box 1802
Hartford, CT 06144
Phone: 203-233-8134

N2N Squared Computing
5318 Forest Ridge Rd.
Silverton, OR 97381

Oasis Systems
2765 Reynard Way
San Diego, CA 92103
Phone: 714-291-9489

Offete Enterprises Inc.
1306 South B St.
San Mateo, CA 94402

OHM Electronics
746 Vermont St.
Palatine, IL 60067

Omega Microware, Inc.
222S. Riverside Plaza
Chicago, IL 60606
Phone: 312-648-1944

Omegasoft
Box 70265
Sunnyvale, CA 94086
Phone: 408-733-6979

Omegasoft
P.O. Box 842
Camarillo, CA 93010
Phone: 805-987-6426

Omnico
3300 Buckeye Rd.
Atlanta, GA 30341
Phone: 404-455-8460

Ongoing Ideas
RD #1 Box 20
Starksboro, VT 05487
Phone: 805-453-4442

On Line Systems
36575 Mudge Ranch Rd.
Coarsegold, CA 93614
Phone: 209-683-6858

Opportunities For Learning Inc.
8950 Lurline Ave.
Chatsworth, CA 91311
Phone: 213-341-2535

Optimized Systems Software
10389-D Landsdale Ave.
Cupertino, CA 95014
Phone: 408-446-3099

Options 80
Box 471
Concord, MA 01742

Orange Computers Inc.
10-1495 Bonhill Road
Mississauga, Ontario 15T1M2
Phone: 416-677-7782

Orange Micro Inc.
3150 East La Palma
Anaheim, CA 92806
Phone: 714-630-3620

Organic Software
1492 Windsor Way
Livermore, CA 94550
Phone: 415-455-4034

Orion Business Systems
11777 Bernardo Plaza Court
San Diego, CA 93614
Phone: 714-485-8580

Orion Software Associates
147 Main Street
Ossining, NY 10562
Phone: 914-726-4903

Osborne/McGraw-Hill
630 Bancroft Way
Berkeley, CA 93614
Phone: 415-548-2805

Output Inc.
Box 519
Plymouth, MI 48170
Phone: 313-397-1633

Overdrive Computer Corporation
1131 University Blvd. W. S-2102
Silver Spring, MD 20902
Phone: 301-649-5359

Pacific Bancorporation Data Systems Inc.
Box 6008
Bakersfield, CA 93386
Phone: 805-395-3231

Pacific Exchanges
100 Foothill Blvd.
San Luis Obispo, CA 93401
Phone: 800-235-4137

Padmede Computer Services
351 Fleet Road
Hampshire, England

Pakre Inc.
5230 N. Clark Suite 5
Lakewood, CA 90712

Pansophics Ltd.
Whistle Stop Mall
Rockport, MA 01966

Passport Designs Inc.
785 Main Street
Half Moon Bay, CA 94019
Phone: 415-726-0280

Paul's Electric Computer
Box 74157
Los Angeles, CA 90004

PBL Corp.
Box 559
Wyzata, MN 55319
Phone: 612-473-3769

PCD Systems
Box 143
Penn Yan, NY 14527
Phone: 315-536-3734

Peachtree Software, Inc.
3445 Peachtree Rd. N.E.
Atlanta, GA 30326
Phone: 404-239-3000

Pear Software
407 Terrace
Ashland, OR 97520
Phone: 503-482-8122

Peelings II Inc.
2260 Cleander
Las Cruces, NM 88001

Penguin Software
830 4th Ave.
Geneva, IL 60134
Phone: 312-232-1984

People's Computer Company
1263 El Camino Real
Menlo Park, CA 94025
Phone: 415-323-3111

Peripherals Plus
39 East Hanover Ave.
Morris Plains, NJ 07950
Phone: 800-631-8112

Persimmon Software
502C Savannah St.
Greensboro, NC 27406
Phone: 919-275-5824

Personal Business Systems
4306 Upton Ave. South
Minneapolis, MN 55410
Phone: 612-929-4120

Personal Computer Center
3819 West 95th St.
Overland Park, KS 66206
Phone: 913-381-1155

Personal Computer Products Inc.
16776 Bernardo Center Drive
San Diego, CA 92128
Phone: 619-485-8411

Personal Computer Service
1515 Andrews Drive
Las Cruces, NM 88001
Phone: 505-522-5543

Personal Software Inc.
1330 Bordeaux Drive
Sunnyvale, CA 94086
Phone: 408-745-7841

Peter Hohenbrink Software Co.
P.O. Box 866
Placentia, CA 92670
Phone: 714-993-4441

Phase One Systems Inc.
7700 Edgewater Dr.
Oakland, CA 94621
Phone: 415-438-4850

Phoenix Software, Inc.
64 Lake Zurich Dr.
Lake Zurich, IL 60047
Phone: 312-438-4850

Physcians Practice Management
1810 S Lynhurst Suite Q
Indianapolis, IN 46241

Physical Sciences Inc.
Research Park
Andover, MA 01810
Phone: 617-475-9030

Piccadilly Software
89 Summit Ave.
Summit, NJ 07901

Pickham Software
312 S. Los Angeles St.
Los Angeles, CA 90013
Phone: 213-687-9530

Pirates Harbor Inc.
P.O. Box 8928
Boston, MA 02114
Phone: 617-227-7760

Plain and Simple Software
9003 Lexington N.E.
Albuquerque, NM 87112
Phone: 505-293-2240

Popular Computing Inc.
375 Cochituate Rd.
Framingham, MA 01701
Phone: 800-343-6474

Powersoft, Inc.
Box 157
Pitman, NH 08071
Phone: 609-589-5500

Practical Peripherals Inc.
31245 La Baya Drive
Westlake Village, CA 91362
Phone: 213-991-8200

Practical Software Ltd.
P.O. Box 3000
Pomona, NY 10970

Prentice Corporation
266 Caspain Drive
P.O. Box 3544
Sunnyvale, CA 94088-3544
Phone: 408-734-9810

Prentice-Hall Inc.
200 Old Tappan
Old Tappan, NJ 07675
Phone: 201-767-5000

Princeton Graphics
195 Nassau Street
Princeton, NJ 08540

PrintaColor Corporation
5965 Peachtree Corners East
Norcross, GA 30071
Phone: 404-448-2675

Printers Computer Services
770 S. Brea Blvd. Suite 210
Brea, CA 92621
Phone: 714-990-3547

Pro Bike Inc.
5020 N.W. 63rd
Oklahoma City, OK 73132
Phone: 405-721-6707

Processing Innovations Inc.
10471 S. Brookhurst
Anaheim, CA 92804
Phone: 714-535-8161

Professional Business Software
119 Fremont
San Francisco, CA 94105
Phone: 415-546-1596

Professional Computer Systems
318A Lincoln Court
Bloomingdale, IL 60108
Phone: 312-351-8817

Professional Medical Software
3604 Foothill Blvd.
La Crescenta, CA 91214
Phone: 213-248-2884

Professional Software Technology Inc.
55 Wheeler St.
Cambridge, MA 02138
Phone: 617-497-2077

Program Design Inc.
11 Idar Court
Greenwich, CT 06830
Phone: 203-792-8382

Programmers Software Exchange
2110 N. 2nd St.
P.O. Box 199
Cabot, AR 72023

Programming International
505 Hamilton Ave. Suite 107
Palo Alto, CA 94301
Phone: 415-324-3730

Programs for Learning
Box 954
New Milford, CT 06776
Phone: 203-355-3452

Progressive Software
Box 273
Plymouth Meeting, PA 19432

Prometheus Products Inc.
45277 Fremont Blvd.
Fremont, CA 94538
Phone: 415-490-2370

Pro/Pac Inc.
14925 Memorial Drive Suite 105
Houston, TX 77079
Phone: 713-496-1179

Puget Sound Program Library Exchange
304 Main Ave. Suite 300
Benton, WA 98055
Phone: 206-932-6588

Pygmy Programming
Box 33070
Scottsdale, AZ 85257

Quad Systems
P.O. Box 260279
Tampa, FL 33685

Quadram Corporation
4357 Park Drive
Norcross, GA 30093
Phone: 404-923-6666

Quality Education Designs
Box 12486
Portland, OR 97212
Phone: 503-282-4906

Quality Software
6660 Reseda Blvd. Suite 105
Reseda, CA 91335
Phone: 213-344-6599

Quark Engineering
1433 Williams Suite 11
Denver, CO 80218
Phone: 303-399-1096

Quelo
843 NW 54th
Seattle, WA 98107
Phone: 206-784-8018

Quinsept Inc.
Box 216
Lexington, MA 02173
Phone: 617-862-0404

Qume Corporation
2150 Qume Dr.
San Jose, CA 95131
Phone: 408-942-4000

Racal-Vadic
1525 McCarthy Blvd.
Milpitas, CA 95035
Phone: 408-946-2227

R. C. Electronics Inc.
5386-D Hollister Ave.
Santa Barbara, CA 93111
Phone: 805-968-6614

Radio Shack
(Division of Tandy Corp.)
1800 One Tandy Center
Fort Worth, TX 76102
Phone: 817-390-3011

Raff Craft
Box 1754
Stillwater, OK 74074

Rainbow Computing
Golden Plaza Shopping Center
9719 Reseda Blvd.
Northridge, CA 91324
Phone: 213-349-5560

Rainbow Marketing
3111 Berkshire Road
Baltimore, MD 21214
Phone: 301-426-6812

R-Alpha Software
Box 3332
Crofton, MD 21114
Phone: 301-261-3749

Rana Systems
20620 S Leapwood Ave.
Carson, CA 90746
Phone: 213-583-2353

Random House School Division
400 Hahn Rd.
Westminster, MD 21157
Phone: 800-241-6402

RCI Marketing
19517 Business Center Drive
Northridge, CA 91324

RDA Systems
Box 1456
Lafayette, CA 94549
Phone: 415-283-0573

Reactive Systems Inc.
40 N. Van Brunt St.
Englewood, NJ 07631
Phone: 201-568-0481

Reader's Digest Edctnl. Div.
Pleasantville, NY 10507
Phone: 914-769-7000

Real Estate Microcomputer Systems
2811 N.W. Grant St. Suite C
Corvallis, OR 97330
Phone: 503-757-8887

Real Estate Software
1450 W. Georgia Street
Vancouver, B.C. V6G 2T8
Canada
Phone: 604-669-2262

Reality Automation Inc.
221 N. Lois
La Habra, CA 90631

Realty Software
1116 8th St. Suite F
Manhattan Beach, CA 90266
Phone: 213-372-9419

Reed Holdings Inc.
221 Columbus Ave.
Boston, MA 02116

RE/International Systems Corporation
6404 Wilshire Blvd.
Los Angeles, CA 90048
Phone: 213-852-1054

Renaissance Technology Corp.
1070 Shary Circle
Concord, CA 94518

Reston Publishing Co
11480 Sunset Hills Rd.
Reston, VA 22090
Phone: 800-336-0338

Reston Software
11480 Sunset Hills Rd.
Reston, VA 22090
Phone: 800-336-0338

RGB Designs
3375 Woodward Avenue
Santa Clara, CA 95050
Phone: 408-748-0400

R.H. Electronics
566 Irelan Street
Buelton, CA 93427
Phone: 805-688-2047

Rhino Robots Inc.
308 S. State St.
Champaign, IL 61920

Richard Adams
894 W. 11th
Eugine, OR 97402

Richard Lorance & Assoc Inc.
3336 N. 32nd St.
Phoenix., AZ 85018
Phone: 602-957-4670

Right On Programs
Box 997
Huntington, NY 11743

Rising Sun Software
P.O. Box 11020
Oakland, CA 94611
Phone: 415-482-3391

Ritam Corp.
Box 921
Fairfield, IA 52556
Phone: 515-472-8262

Riverbank Software
Box 128
Denton, MD 21629
Phone: 301-479-1312

RKS Enterprises Inc.
643 South 6th Street
San Jose, CA 95112
Phone: 408-288-5565

RKS Industries Inc.
4865 Scotts Valley Dr.
Scotts Valley, CA 95066

RMI Media Products Inc.
120 W. 72nd St.
Kansas City, MO 94114
Phone: 800-821-5480

Robec Inc.
Route 309
Montgomeryville, PA 18936
Phone: 215-628-4960

Rocky Mountain Software
1038 Hamilton St.
Vancouver, B.C. V6B 2R9
Canada
Phone: 604-681-3371

Rogers Products Company Inc.
249-B East Emerson Ave.
Orange, CA 92665
Phone: 714-974-6302

Roklan Corporation
3335 N. Arlington Heights
Arlington, IL 60004
Phone: 312-392-2525

Rosen Grandon Associates
7807 Whittier St.
Tampa, FL 33167
Phone: 813-985-4911

RTR Software, Inc.
444 Executive Center Blvd. Suite 225
El Paso, TX 79902
Phone: 915-544-4397

Rixon Inc.
2120 Industrial Pkwy.
Silver Spring, MD 20904
Phone: 301-622-2121

S & H Software
58 Van Orden
Harr Park, NJ 07640

S Richard Slade Creative Mrktg.
P.O. Box 484
Santa Barbara, CA 93102

Salba Software
206 E.Cypress Ave.
Elmwood, IL 61529

Sams Books and Software
4300 West 62nd Street
P.O. Box 7092
Indianapolis, IN 46206

Samurai Software
P.O. Box 5515 Dept A-1
Chula Vista, CA 92012-5515
Phone: 619-691-0609

San Juan Unified School Dist.
6141 Sutter Dr.
Carmichael, CA 95608
Phone: 916-944-3614

Sandow Enterprises
4825 Geer Road
Hughson, CA 95326
Phone: 209-883-0665

Santa Clara Systems Inc.
560 Division St.
Campbell, CA 95008

SAS Electronics
3091 North Bay Dr.
North Bend, OR 97459

Sat.Trac International
4543 Templeton Gap Rd
Colorado Springs, CO 80909
Phone: 303-574-4150

Saturn Systems
Box 8050
Ann Arbor, MI 48107
Phone: 313-665-6416

S-C Software Corp.
2331 Gus Thomasson Suite 125
Box 280300
Dallas, TX 75228

Scelbi Publications
P.O. Box 232
Seymour, CT 06483

Scharf Software
Box 18445
Irvine, CA 92713
Phone: 714-557-9206

Scholastic Inc.
730 Broadway
New York, NY 10003
Phone: 212-505-3000

School Courseware Journal
4919 N. Millbrook #222B
Fresno, CA 93726
Phone: 209-225-0953

School Microware Dresden Associates
Box 256
Dresden, MN 04342
Phone: 207-737-4466

School Office Software Systems
3408 Dover Road
Durham, NC 27707

Science Research Associates Inc.
155 N. Wacker Dr.
Chicago, IL 60606
Phone: 800-621-0664

Scientific Software
Box 208
Wausau, WI 54401
Phone: 715-845-2066

Scitronics Inc.
523 S. Clewell St.
P.O. Box 5344
Bethlehem, PA 18015
Phone: 215-868-7200

Scott Foresman & Co. Electronics Publishing
1900 East Lake Ave.
Glenview, IL 60025
Phone: 312-729-3000

Scott Instruments
1111 Willow Springs Dr.
Denton, TX 76201
Phone: 817-387-9514

Scripps Data Systems Inc.
9747 Business Park Ave Suite 2020
San Diego, CA 92131
Phone: 714-695-1540

SEI
Box 400
Big Flats, NY 14841
Phone: 607-562-3287

Select Information Systems
919 Sir Francis Drake Blvd.
Kentfield, CA 94904
Phone: 415-459-4003

Selectone Corporation
Computer Product Division
28301 Industrial Blvd.
Hayward, CA 94545

Sensational Software
39 East Hanover Ave.
Morris Plains, NJ 07950
Phone: 800-631-8112

Sensible Software
6619 Perham Dr.
W. Bloomfield, MI 48033
Phone: 313-977-3869

Sentient Software
Box 4929
Aspen, CO 81612
Phone: 303-925-9293

Serendipity Systems Inc.
419 W. Seneca St.
Ithaca, NY 14850
Phone: 687-277-4889

Serra Software
536 Sahara Ave.
Las Vegas, NV 89104
Phone: 701-647-2673

Serus Software
1537 Howe Ave.
Sacramento, CA 95825
Phone: 916-920-8981

Services Unique Inc.
2441 Rolling View Drive
Dayton, OH 45401
Phone: 513-429-0696

Sevedge Software
5101 N. Kostner St.
Chicago, IL 60630

SGC
342 Quartz Circle
Livermore, CA 94550
Phone: 415-490-3420

Shafer Software
465 S. Mathilda Ave. Suite 202
Sunnyvale, CA 94086
Phone: 408-730-0179

Shanali Inc.
171 S.Livingston
Livingston, NJ 07039
Phone: 201-992-7971

Sharti Systems Inc.
Box 94543
Shaumberg, IL 60194

Sherman Electronics
Box 63-04
Miami, FL 33163
Phone: 305-948-8000

Shoe String Software
1235 Candlelight
Houston, TX 77018

Sierra On-Line Inc.
36575 Mudge Ranch Road
Coarsegold, CA 93614
Phone: 209-683-6858

Sierra Software
563 E. Sahara Ave.
Las Vegas, NV 89104
Phone: 702-649-1083

Silicon Valley
652 Bair Island Road
Redwood City, CA 94063
Phone: 415-364-0752

Silicon Valley Systems Inc.
1625 El Camino Real
Belmont, CA 94002
Phone: 415-593-4344

Simple Soft Inc.
480 Eagle Dr. Suite 101
Elk Grove, IL 60007
Phone: 312-364-0752

Simulation Software
6035 N. Maplewood Ave.
Chicago, IL 60659

Single Source Solution
2699 Clayton Rd.
Concord, CA 94519

Sir-Tech Software, Inc.
6 Main St.
Ogdensburg, NY 13699
Phone: 315-393-6633

Sirius Software
10364 Rockingham Dr.
Sacramento, CA 95827
Phone: 916-366-1195

Skyline Computing Co.
518 Boston Post Rd.
Orange, CT 06477

Sliwa Enterprises
P.O. Box 7266
Hampton, VA 23666

Small Business Computer Systems
4140 Greenwood
Lincoln, NE 68504
Phone: 402-467-1878

Small Systems Group
Folcroft Park
Folcroft, PA 19032
Phone: 800-345-8102

Smart Systems Incorporated
499 Sulky Lane
Frederick, MD 21701
Phone: 301-694-8307

Smart Ware
2281 Cobble Stone Court
Dayton, OH 45431
Phone: 513-426-3579

Smith Micro Software
Box 604
Sunset Beach, CA 90742
Phone: 213-592-1032

Snave Systems
P.O. Box 957
Niles, IL 60648-0957
Phone: 312-966-4505

Society for Visual Education Inc.
1345 Diversay Parkway
Chicago, IL 60614
Phone: 312-525-1500

Sof/Sys Inc.
4306 Upton Ave. South
Minneapolis, MN 55410

Soft Control Systems
Box 599
West Milford, NJ 07480
Phone: 201-728-8750

Soft One
315 Dominion Dr.
Newport News, VA 23603
Phone: 804-877-6046

Soft Touch
P.O. Box 7200
Costa Mesa, CA 92626
Phone: 714-645-1019

Softagon
Box 774M
Morristown, NJ 07960
Phone: 201-539-3770

Softalk Publishing Inc.
11201 Magnolia Blvd.
North Hollywood, CA 91601
Phone: 213-985-5763

Softape
10432 Burbank Blvd.
North Hollywood, CA 91602
Phone: 213-985-5763

Softart Software
46 Green Street
Carver, MA 02330
Phone: 617-866-2103

Softcorp International
229 Huber Village Blvd.
Westerville, OH 43081

Softech
144 W. 15th St.
North Vancouver, B.C. V7M
Canada
Phone: 604-984-0477

Softech Microsystems
9494 Black Mountain Rd.
San Diego, CA 92126
Phone: 714-578-6105

Softhouse
Box 6383
Rochester, CA 55901
Phone: 507-285-9121

Softpak Associates
26 Venice Blvd.
Marina Del Rey, CA 90291
Phone: 213-822-1830

Softraders International
1610 Shoemaker Dr.
Murphysboro, IL 62966

Softsel
8295 S. La Cienega Blvd.
Inglewood, CA 90301
Phone: 800-421-5770

Softsmith
Box 272
Kansas City, MO 64141
Phone: 913-492-8882

Softstalker
P.O. Box 689
1376 Overlook Drive
Steamboat Springs, CO 80477
Phone: 800-552-1500

Softswap
333 Main St.
Redwood City, CA 94063
Phone: 415-363-5472

Softwand Software
3125 Vendome Terrace
Norfolk, VA 23509

Software Alternatives Inc.
4841 Monroe St. Suite 101-B
Toledo, OH 43623
Phone: 419-472-2122

Software Arts Inc.
27 Mica Lane Rd.
Wellesley, MA 02181
Phone: 617-491-2100

Software By H Inc.
P.O. Box 6592
Rochester, MN 55901

Software Connections
1800 Wyatt Dr. Suite 1
Santa Clara, CA 95054
Phone: 408-988-3704

Software Creations
10720 S. Hoover
Los Angeles, CA 90044
Phone: 213-754-4431

Software Emporium
Box 3232
Tulsa, OK 74101
Phone: 918-662-7314

Software Dimensions Inc.
6371 Auburn Blvd.
Citrus Heights, CA 95610
Phone: 916-722-8000

Software Factory, The
23849 La Salle Canyon Dr.
Newhall, CA 91321
Phone: 805-259-8558

Software House Inc.
695 E. 10th North
Logan, UT 84321
Phone: 800-453-2710

Software Hows
P.O. Box 36275
Los Angeles, CA 90036
Phone: 213-731-0876

Software Master
3330 Hillcroft #BB
Houston, TX 77057
Phone: 713-266-5771

Software Productions Inc.
2357 Southway Dr.
P.O. Box 21341
Columbus, OH 43221
Phone: 614-486-3563

Software Publishing Corp.
2021 E. Landings Dr.
Mountain View, CA 94043
Phone: 415-962-8911

Software Resources
44 Brattle Street
Cambridge, MA 02138
Phone: 617-497-5900

Software Resources Computer Works Inc.
789 E. Market
Harrisonburg, VA 22801

Software Solutions
9124 Highway 17
Scotts Valley, CA 95066
Phone: 408-438-2433

Software Sorcery Inc.
7927 Jones Branch Dr.
McLean, VA 22102
Phone: 703-385-2944

Software Support Inc.
One Edgell Road
Framingham, MA 01701
Phone: 617-872-9090

Software Technology for Computers
Box 428
Belmont, MA 02178
Phone: 617-923-4334

Software To Go
1948 Oak Ridge Turnpike
Oak Ridge, TN 37830
Phone: 615-482-9592

Software Unlimited
P.O. Box 6361
Clearwater, FL 33518

Softworks Inc.
607 Wellington Rd.
Chicago, IL 60657
Phone: 312-327-7666

Solarsoft Inc.
Box 124
Snowmass, CO 81654
Phone: 303-927-4411

Solartek
Box 298
Guilderland, NY 12084

Solidus International Corp.
#204 4202 Guide Meridi
Bellingham, WA 98226
Phone: 206-734-3744

Solution Inc.
97 College St. Box 989
Montpelier, VT 05602

Sonora Computing
2300 Candelaria Rd.
N.E. Albuquerque, NM 87107

Sophisticated Microsystems Inc.
550 Frontage Rd.
Northfield, IL 60093
Phone: 312-441-5802

Sophisticated Software
650 Foothill Blvd.
La Canada, CA 91011

Sorcim
2310 Lundy Ave.
San Jose, CA 95131
Phone: 408-942-1727

Sorrento Valley Associates
11722 Sorrento Valley Road
San Diego, CA 92121
Phone: 714-452-0101

Sound Software Systems
Box 177
Tacoma, WA 98401
Phone: 206-759-3526

Source Telecomputing Corp.
1616 Anderson Rd.
McLean, VA 22102
Phone: 703-734-7500

South Texas Software
P.O. Box 34091
San Antonio, TX 78233
Phone: 512-654-7781

Southeastern Software
7743 Briarwood Dr.
New Orleans, LA 70127
Phone: 504-246-7937

Southern California Research
Box 2231
Goleta, CA 93118
Phone: 805-685-9131

Southern Center for Research & Innovation
P.O. Box 1713
Hattiesburg, MS 39403

Southern Micro Systems
P.O. Box 1981
Burlington, NC 27216
Phone: 919-226-7610

Southwest Edpsych Services Inc.
P.O. Box 1870
Phoenix, AZ 85001
Phone: 602-253-6528

Southwestern Data Systems
10761-E Woodside Ave.
Santee, CA 92071
Phone: 714-562-3670

Southwind Computer Sales
Box 242A
Dundee, NY 14837

Space-Time Associates
20-39 Country Club Drive
Manchester, NH 03102
Phone: 603-625-1094

Sparrow Distributors
Canoga Park, CA 91304

Specialty Software
7403 W. Canal Drive
Suite P-396
Kennewick, WA 99336

Spectra Soft Inc.
P.O. Box 277
Chandler, AZ 85224

Spectrum Software
142 Carlow Box 20
Sunnyvale, CA 94086
Phone: 408-738-4387

Spider Software
98 Avondale Rd.
South Croydon, Surrey
England

Spies Laboratories
Box 336
Lawndale, CA 90260
Phone: 213-644-0056

Spinnaker Software
26 Brighton St.
Belmont, MA 02178
Phone: 617-484-8444

Sport Tech
350 Union Station Building
Kansas City, MO 64108
Phone: 816-474-0860

Sportsware Inc.
1700 Stumpf Suite 703
Gretna, LA 70053

SSM Microcomputer Products
2190 Paragon Dr.
San Jose, CA 95131
Phone: 408-946-7400

S.S.R. Corp.
1600 Lyell Ave.
Rochester, NY 14606
Phone: 716-254-3200

STB Systems Inc.
1701 N. Greenville 703
Richardson, TX 75081
Phone: 214-234-8750

Stellation Two
Box 2342
Santa Barbara, CA 93120
Phone: 805-966-1140

Step One Inc.
1870 N. Sheldon St.
St Paul, MN 55113

Sterling Swift Publishing Co.
Box 188
Manchaca, TX 78562
Phone: 512-444-7570

Stiennon Radiology Group
One South Park St 220
Madison, WI 53715

Stock Market Software Inc.
44 Front St.
Ashland, MA 01721
Phone: 617-881-5190

Stonehenge Computer Company
89 Summit
Summit, NJ 07901
Phone: 414-454-6500

Stoneware Inc.
50 Belvedere St.
San Rafael, CA 94901
Phone: 415-454-6500

Stoneware Microcomputer Products
See Stoneware Inc.

Strategic Simulations
465 Fairchild Suite 108
Mountain View, CA 94043
Phone: 415-964-1353

Strawberry Tree Computers
949 Cascade Dr.
Sunnyvale, CA 94087
Phone: 408-736-3083

Jack Strick & Associates
949 S. Southlake Drive
Hollywood, FL 33019
Phone: 305-925-7004

Strictly Soft Ware
P.O. Box 338
Granville, OH 43023
Phone: 614-587-2938

Strobe Inc.
897-5A Independence Ave.
Mountain View, CA 94043
Phone: 415-969-5130

Sublogic Communications Corp.
713 Edgebrook Dr.
Champaign, IL 61820
Phone: 217-359-8482

Sugar Hollow Shapes
120 Terrybrook Dr.
Charlottesville, VA 22901
Phone: 804-823-2292

Sun Research Inc.
Box 210
New Durham, NH 03855
Phone: 603-859-7110

Sunnyside Soft
5815 E. Parkside Dr.
Fresno, CA 93727

Super Score
Box 37
Saugus, CA 91350
Phone: 805-255-5850

Superior Software Inc.
4312 Arizona Ave.
Kenner, LA 70062
Phone: 504-468-2273

Supersoft Associates
P.O. Box 1628
Champaign, IL 61820
Phone: 217-359-2112

Supply 21
230 Woodmont Rd.
P.O. Box 3176
Milford, CT 06460
Phone: 203-878-5797

SVE-Society for Visual Education Inc.
1345 W. Diversey Parkway
Chicago, IL 60614
Phone: 312-525-1500

Sweet Micro Systems
150 Chestnut Street
Providence, RI 02903
Phone: 401-273-5333

Sybex Inc.
2344 Sixth St.
Berkeley, CA 94710
Phone: 415-848-8233

Sympathetic Software
9531 Telhan Dr.
Huntington Beach, CA 92646

Symtec Inc.
15933 W. Eight Mile Road
Detroit, MI 48235
Phone: 313-272-2950

Synergistic Software
830 N. Riverside Dr. Suite 201
Renton, WA 98055
Phone: 206-226-3216

Synetex Industries
15050 N.E. 95th Street
Redmond, WA 98052

Synetix Systems Inc.
15050 N.E. 95th Street
Redmond, WA 98052

Syntauri
3506 Waverley St
Palo Alto, CA 94306
Phone: 415-494-1017

Syntax Corporation
4500 W. 72nd Terrace
Prairie Village, KS 66208
Phone: 913-362-9667

Syntonic Software Corp.
10635 Richmond
Houston, TX 77042
Phone: 800-231-5768

System Decisions Group
149 Rowayton Ave.
Rowayton, CT 06853

System Design Associates Inc.
723 Kanawha Blvd. East
Charleston, WV 25301

System Design Lab
2612 Artesia Blvd. Suite B
Redondo Beach, CA 90278
Phone: 213-374-4417

Systems Fabricators
736 Hermosa Ave.
Hermosa Beach, CA 90254
Phone: 213-372-6273

Systems Formulate Corp.
39 Town & Country Village
Palo Alto, CA 94301
Phone: 415-969-7499

Systems Plus
3975 East Bayshore
Palo Alto, CA 94304
Phone: 415-969-7047

Systems Support
4 Granite Street
Braintree, MA 02184
Phone: 617-843-9755

T & W System Inc.
19437 Mt. Langley Suite B
Fountain Valley, CA 92708
Phone: 714-963-3913

Tackaberry Software
P.O. Box 2857
Ormond Beach, FL 32074
Phone: 904-677-2054

Tallgrass Technologies
9207 Cody
Overland Park, IL 60035
Phone: 913-492-6002

Tape'n Text-Williamsville Pub.
113 Central Ave.
Fredonia, NY 14063

Target Enterprises
50 Arvesta St.
Springfield, MA 01118

TCS Software
3209 Fondren
Houston, TX 77063
Phone: 713-977-7505

Teach Yourself By Computer (TYC) Software
40 Stuyvesant Manor
Geneseo, NY 14454
Phone: 716-234-3005

Tebbs Techniques
P.O. Box 817
Pleasant Hill, OR 97455

Tech Designs Inc.
3638 Grosvenor Dr.
Ellicott City, MD 21043
Phone: 301-291-0818

Technical Software
Box 73043
Metairie, LA 70033
Phone: 504-443-1637

Technologies Inc.
3204 Monroe St.
Rockville, MD 20852
Phone: 301-984-3199

Technology Information Corp
22 Central Ave. Suite F
Boulder, CO 80301

Technology Publishing Co.
38 Commercial Wharf
Boston, MA 02110
Phone: 617-227-4700

Teck Associates
Box 8732
White Bear Lake, MN 55110
Phone: 612-535-5330

Tecmar, Inc.
23600 Mercantile Rd.
Cleveland, OH 44122
Phone: 216-464-7410

Telephone Software Connection
Box 6548
Torrance, CA 90504
Phone: 213-516-9430

Telesphere Corp., The
P.O. Box 2443
Secaucus, NJ 07094

Telesys Group Inc., The
5455 Winsborne Court
Columbia, MD 21045

Tele-Terminals Inc.
7216 Boone Ave.
Brooklyn Park, MN 55428
Phone: 612-535-5330

Terrapin Inc.
380 C Green St.
Cambridge, MA 02139
Phone: 617-492-8816

Texcom
Box 24472
Houston, TX 77013
Phone: 713-458-3720

TG Products
1104 Summit Ave. Suite 106
Plano, TX 75074
Phone: 214-424-8568

T.H.E.S.I.S.
Box 147
Garden City, MI 48135
Phone: 313-595-4722

The Computer Stop
16919 A Hawthorne Blvd.
Lawndale, CA 90260
Phone: 213-371-4010

Third Millennium Engineering Corp.
1015 Gayley Avenue Suite 394
Los Angeles, CA 90024

Thunderware Inc.
Box 13322
Oakland, CA 94661
Phone: 415-652-1737

Ties Minnesota School Dist.
1925 W County Road B2
St Paul, MN 55113

Tiger Publications
Box 8359
Newport Beach, CA 92660
Phone: 714-296-2115

Time Proven Systems Inc.
1210 Sheppard Ave. East Suite 101
Willowdale, Ontario M2K 1EK
Canada
Phone: 416-491-6629

TIMECOR
Four Longfellow Place
P.O. Box 8928
Boston, MA 02114
Phone: 617-720-4090

TMQ Software Inc.
82 Fox Hill Drive
Buffalo Grove, IL 60090
Phone: 312-520-4440

TNW
3444 Hancock St.
San Diego, CA 92110
Phone: 714-975-1215

Tovatech
1903 Fordham Way
Mountain View, CA 94040
Phone: 415-968-0392

Trace Systems Inc.
1928-A Old Middlefield
Mountain View, CA 94043
Phone: 415-964-3115

Transtar
Box C-96975
Bellevue, WA 98009

Trilobyte Software Inc.
7820 Exeter St.
New Orleans, LA 70126
Phone: 504-733-3195

Trinebula Inc.
Computer Products Division
P.O. Box 45191
Tacoma, WA 98445

T.S. Denison & Co. Inc.
9601 Newton Avenue South
Minneapolis, MN 55431
Phone: 612-888-1468

TSK Electronics Corporation
18005 Cortney Court
City of Industry, CA 91748

T-Track Systems Inc.
9025 Wilshire Blvd.
Beverly Hills, CA 90211
Phone: 213-274-0804

Tucker Farm Consultants
Box 756
Willernie, MN 55090
Phone: 612-426-2791

Turnabout Enterprises
P.O. Box 5028H
Mishawaka, IN 46545
Phone: 219-256-5967

Turnkey Software
13078 Mindanao Way Suite 314
Marina Del Rey, CA 90291

Tymac Controls Corp.
7 Main Street
Franklin, NJ 07146

Unicom Division of United Camera Inc.
297 Elmwood Avenue
Providence, RI 02907
Phone: 401-467-5600

Unidata Inc.
1304 Langham Creek Drive Suite 338
Houston, TX 77084
Phone: 713-492-6356

Unique Information Systems Inc.
30 N. LaSalle Suite 29
Chicago, IL 60602
Phone: 312-346-7933

United Software of America
750 Third Ave.
New York, NY 10017
Phone: 212-687-5001

Univair
10327 Lambert International Airport
St. Louis, MO 63142
Phone: 314-426-1099

Universal Data Systems
5000 Bradford Drive
Huntsville, AL 35085
Phone: 205-837-8100

Universal Systems for Education Inc.
2120 Academy Circle
Colorado Springs, CO 80909
Phone: 303-574-4575

Urban Aggregates Inc.
6431 Brass Knob
Columbia, MD 21044
Phone: 301-997-4251

USA Software
See United Software of America.

USE Electronics
103 Godwin Avenue
Midland Park, NJ 07432

USE Inc.
2120 Academy Circle
Colorado Springs, CO 80909
Phone: 303-574-4575

USI International
71 Park Lane
Brisbane, CA 94005
Phone: 415-468-4900

Utilities Engineering
Box 299
Brigantine, NJ 08203
Phone: 609-266-1774

Van Nostrand Reinhold
7625 Empire Dr.
Florence, KY 41042
Phone: 606-525-6600

Vanguard Systems Corp.
6901 Blanco
San Antonio, TX 78216
Phone: 512-340-1978

V.E.C.T
160 Paul Street
Oconto Falls, WI 54154

Vector Electronic Company
12460 Gladstone Ave.
P.O. Box 4336
Sylmar, CA 91342
Phone: 213-365-9661

Ven-Tel Inc.
2342 Walsh Ave.
Santa Clara, CA 95051
Phone: 408-727-5721

Ventura Educational Systems
2239 Townsgate Road
Westlake Village, CA 91361
Phone: 805-496-2247

Venture Microware Ltd.
17125 Rt. 5W
East Dubuque, IL 61025
Phone: 815-747-6971

Verba Gloria
802 12th Avenue Suite B
Menomonie, WI 54751
Phone: 715-235-3126

Vernier Software
2920 SW 89th Street
Portland, OR 97225

Vernon Tech Corp.
Box 215
Springtown, PA 18081
Phone: 215-346-7757

Versa Computing Inc.
3541 Old Conejo Rd. #104
Newbury Park, CA 91320
Phone: 805-498-1956

Video Associates Labs Computer Division
2304 Hancock Dr. 1F
Austin, TX 78756

Video World
2224 University
Peoria, IL 61604

Videx, Inc.
897 N.W. Grant Ave.
Corvallis, OR 97330
Phone: 503-758-0521

Village Software
31220 La Baya Dr
Westlake Village, CA 91362
Phone: 805-946-9000

Virtual Combinatics
P.O. Box 755
Rockport, MA 01966
Phone: 617-546-6553

VisiCorp
2895 Zanker Rd.
San Jose, CA 95131
Phone: 408-946-9000

Vista Computer Company
1317 E. Edinger Avenue
Santa Ana, CA 92705
Phone: 714-953-0523

Vital Informations
7899 Mastin Dr.
Overland Park, KS 66204
Phone: 800-255-5119

Vitalograph Medical Instrumentation
8347 Quivira
Lenexa, KS 66215
Phone: 800-255-6626

VK Utilities
4147-0 California Ave. S.W.
Seattle, WA 98116

VMI Visual Material Inc.
4172 Grove Ave.
Gurnee, IL 60031
Phone: 312-249-1710

Voice Machine Communications Inc.
1000 S. Grand Avenue
Santa Ana, CA 92705
Phone: 714-639-6150

Voicetek
P.O. Box 388
Goleta, CA 93116
Phone: 805-685-1854

Votrax
500 Stephenson Hwy
Troy, MI 48084
Phone: 313-588-2050

Voyager Software
147 22nd Ave.
San Francisco, CA 94115

VRdata
777 Henderson Blvd. N-6
Folcroft, PA 19032
Phone: 800-345-8102

Vulcan Software
1805 Saulter Road
Birmingham, AL 35209
Phone: 205-871-5510

Vynet Corp.
2405 Qume Dr.
San Jose, CA 95131
Phone: 408-942-1037

W. Helm
827 Vinton Court
Thousand Oaks, CA 91360
Phone: 805-497-9041

Wadsworth Electronic Publishing
10 David Dr.
Belmont, CA 94002

Wallace Micro Mart
2619 N. University
Peoria, IL 61604
Phone: 309-685-7876

Wang Laboratories
One Industrial Ave.
Lowell, MA 01851
Phone: 617-459-500

Warner Computer Systems Inc.
605 3rd Ave.
New York, NY 10158
Phone: 212-986-1919

Waverly Press
428 E. Preston St.
Baltimore, MD 21202

Wayne Green Inc.
80 Pine St.
Peterborough, NH 03458
Phone: 603-924-9471

Webster McEvoy & Associates
4 Embarcadero Center S.
San Francisco, CA 94111
Phone: 415-956-8289

West Coast Consultants
1775 Lincoln Blvd.
Tracy, CA 95376

West Side Electronics
Box 636
Chatsworth, CA 91311

Western Datacom
5083 Market Street
Youngstown, OH 44512
Phone: 216-788-6583

Western Micro Distributing
Box 633 Postal Station G
Calgary, Alberta
Canada
Phone: 403-247-1621

Western Microdata Enterprises
Box G33 Postal Station G
Calgary, Alberta T3A 2G1
Canada

Weststar Corp.
2665 Shell Beach Rd. E.
Shell Beach, CA 93449

Westware Software Inc.
2455 S.W. 4th Ave.
Ontario, OR 97914
Phone: 503-881-1477

W.H. Nail Co.
275 Lodgeview Dr.
Oroville, CA 95965
Phone: 916-589-2043

Whitney Educational Services
1499 Bayshore Highway
Burlingame, CA 94010
Phone: 415-570-7017

Wild Video
5245 W. Diversey
Chicago, IL 60639
Phone: 312-622-9606

Wildcat Computing Inc.
1160 Park Boulevard
Plano, TX 75074
Phone: 214-424-3582

William Kaufmann Inc.
95 First St.
Los Altos, CA 94022
Phone: 415-948-5810

Wilson Jones
6150 Touhy Ave.
Niles, IL 60648
Phone: 312-774-7700

Wims Computer Consulting
6723 E. 66th Place
Tulsa, OK 74133
Phone: 918-492-9036

Winchendon Group, The
3907 Lakota Rd.
P.O. Box 10114
Alexandria, VA 22310

Window Inc.
469 Pleasant St.
Watertown, MA 02172
Phone: 617-923-9147

Wise Owl Workshop
1168 Avenida De Las Pal
Livermore, CA 94550

Witzel
P.O. Box 2123
Littleton, CO 80161

Word-Power
P.O. Box 736
El Toro, CA 92630
Phone: 714-859-7145

Worlock Software
1900 Emery St. Suite 318
Atlanta, GA 30318
Phone: 404-352-1015

W.R.C. Smith Publication
1750 Peachtree Rd. N.W.
Atlanta, GA 30357

Wyman Associates
421 Seville Way
San Mateo, CA 94402
Phone: 415-345-0380

Xcomp
7566 Trade St.
San Diego, CA 92121
Phone: 714-271-8730

Xebec-An Msc Company
432 Lakeside Drive
Sunnyvale, CA 94086
Phone: 408-735-1340

XPS Inc.
323 York Road
Carlisle, PA 17013
Phone: 800-233-7512

Yucaipa Software
12343 12th St.
Yucaipa, CA 92399
Phone: 714-797-6331

Zeitgeist
5150 N. 6th St Suite 179
Fresno, CA 93710
Phone: 209-222-8244

Zeta Systems Inc.
1725 Adelaide Blvd.
Akron, OH 44305

Zeus Computing
Box 712
Downey, CA 90241

Zircon International Inc.
475 Vandell Way
Campbell, CA 95008
Phone: 408-866-8600

ZIYAD Inc.
100 Ford Road
Denville, NJ 07834

Zork Users Group
Box 20923
Milwaukee, WI 53220

Zweig Associates
1711 McGaw Ave.
Irvine, CA 92714
Phone: 714-549-3246

Zygo Industries
P.O. Box 1008
Portland, OR 97207
Phone: 503-297-1724

3G Company Inc.
Rt. 3 Box 28D
Gaston, OR 97119

3M Data Recording Products Div.
Box 33600
St Paul, MN 55133
Phone: 612-733-9572

20th Century Business Systems
324 W 145th Place
Riverdale, IL 60627
Phone: 312-841-3400

21st Century Tech
5271 Princton
Memphis, TN 38119

6502 Program Exchange, The
2920 W. Moana
Reno, NV 89509

Additional Vendors

Discwasher
1407 North Providence Rd.
P. O. Box 6021
Columbia, MO 65205
Phone: 314-449-0941

Quark Peripherals
2525 West Evans
Suite 220
Denver, CO 80219
Phone: 800-543-7711

Video 7 Inc.
12340 S. Saratoga/Sunnyvale Rd.
Suite 1
Sunnyvale, CA 95070
Phone: 408-725-1433

D

Applesoft Commands

COMMAND	EXAMPLE

ABS — **ABS(−3.14159)**
Returns the absolute value of the argument. The example returns 3.14159.

ASC — **ASC("QUEST")**
Returns the decimal ASCII code for the first character in the argument. In the example, 81 (the ASCII code for Q) will be returned.

ATN — **ATN(2)**
Returns the arctangent, in radians, of the argument. In the example, 1.10714872(radians) will be returned.

CALL addr — **CALL−922**
Causes execution of a machine-language subroutine at the memory location whose decimal address is specified. The example causes a line feed.

CHR$** — **CHR$(65)
Returns the ASCII character that corresponds to the value of the argument, which must be between the values 0 and 255. The example returns the letter A.

CLEAR
Sets all variables to zero and all strings to null, and clears the stack.

COLOR — **COLOR=12**
Sets the color for plotting in low-resolution graphics mode. In the example, color is set to green. Color is set to zero by GR. Color names and their associated numbers are

0 black	**4** dark green	**8** brown	**12** green
1 magenta	**5** grey	**9** orange	**13** yellow
2 dark blue	**6** medium blue	**10** grey	**14** aqua
3 purple	**7** light blue	**11** pink	**15** white

To find the color of a given point on the screen, use the SCRN command.

CONT
If program execution has been halted by STOP, END, CTRL-C, or reset 0G return, the CONT command causes execution to resume at the next instruction (like GOSUB)—not the next line number. Nothing is cleared. After reset 0G return, the program may not CONTinue properly because some program pointers and stacks are cleared. CONT cannot be used if you have:

- modified, added, or deleted a program line.
- gotten an error message after stopping execution.

COS — **COS(2)**
Returns the cosine of the argument, which must be in radians. In the example, −.415146836 is returned.

COMMAND	EXAMPLE

CTRL-C
Can be used to interrupt a RUNning program or a LISTing. It also can be used to interrupt an INPUT if it is the first character entered. The INPUT is not interrupted until the RETURN key is pressed.

CTRL-X
Tells the Apple IIc to ignore the line currently being typed, without deleting any previous line of the same line number. A backslash (\) is displayed at the end of the line to be ignored.

DATA — **DATA JOHN SMITH,"CODE 32",23.45,−6**
Creates a list of elements that can be used by READ statements. In the example, the first element is the literal JOHN SMITH; the second element is the string "CODE 32"; the third element is the real number 23.45; the fourth element is the integer −6. Quotation marks need not be used for literals and strings in data statements.

DEF FN — **DEF FN A(W)=2*W+W**
Allows the user to define one-line functions in a program. First the function must be defined using DEF; later in the program the previously DEFined function may be used. The example illustrates how to define a function FN A(W); it may be used later in the program in the form FN A(23), FN (−7*Q+1), and so on.

FN A(23) causes 23 to be substituted for W in 2*23+23 or 69. Assume Q=2; then FN(−7*Q+1) is equivalent to FN A(−7*2+1) or FN(−13). The function will evaluate to 2*(−13)+(−13),−26−13, or −39.

DEL — **DEL 23,56**
Removes the specified range of lines from the program. In the example, lines 23 through 56 will be DELeted from the program. To DELete a single line, say line 350, use the form DEL 350,350, or simply type the line number and then press the RETURN key.

DIM — **DIM AGE(20,3), NAME$(50)**
When a DIM statement is executed, it sets aside space for the specified arrays with subscripts ranging from 0 through the given subscript. In the example, NAME$(50) will be allotted 50+1 or 51 strings of any length; the array AGE(20,3) will be allotted (20+1)*(3+1), 21*4, or 84 real number elements.

If an array element is used in a program before it is DIMensioned, a maximum subscript of 10 is allotted for each dimension in the element's subscript. Array elements are set to zero when RUN or CLEAR is executed.

DRAW — **DRAW 4 AT 50,100**
Draws shape definition 4 from previously loaded shape table, in high-resolution graphics, starting at x=50,y=100. The COLOR, ROTation, and SCALE of the shape to be drawn must be specified before DRAW is executed.

COMMAND **EXAMPLE**

END
Causes a program to cease execution and returns control to the user. No message is printed.

EXP **EXP(2)**
Returns the value of e raised to the power indicated by the argument to 6 place accuracy, e=2.718289. In the example, 7.3890561 will be returned.

FLASH
Sets the video mode to "flashing," so the output from the computer is alternately shown on the TV screen in white characters on black and then reversed to black characters on a white background. Use NORMAL to return to a non-flashing display of white letters on a black background.

FOR **FOR W=1 TO 20: ... :NEXT W FOR Q=2 TO −3 STEP −2: ... :NEXT Q FOR Z=5 TO 4 STEP 3: ... :NEXT**
Allows you to write a "loop" to perform, a specified number of times, any instructions between the FOR command (the top of the loop) and the NEXT command (the bottom of the loop). In the first example, the variable W counts how many times to do the instructions. The instructions inside the loop will be executed for W equal to 1, 2, 3, ..., 20, then the loop ends (with W=21) and the instruction after NEXT W is executed. The second example illustrates how to indicate that the STEP size as you count will be different from 1. Checking takes place at the end of the loop, so in the third example, the instructions inside the loop are executed once.

FRE(0)
Returns the amount of memory, in bytes, still available to the user. What you put inside the parentheses is unimportant, so long as it can be evaluated by Applesoft BASIC. This command also cleans up free string storage area.

GET **GET ANS$**
Fetches a single character from the keyboard without showing it on the TV screen and without requiring that the RETURN key be pressed. In the example, the typed character is stored in the variable ANS$.

GOSUB **GOSUB 250**
Causes the program to branch to the indicated line (250 in the example). When a RETURN statement is executed, the program branches to the statement immediately following the most recently executed GOSUB.

GOTO **GOTO 250**
Causes the program to branch to the indicated line, 250 in the example.

COMMAND	EXAMPLE

GR

Sets low-resolution graphics mode (40 × 40) for the TV screen, leaving four lines for text at the bottom. The screen is cleared to black, the cursor is moved into the text window, and COLOR is set to 0 (black).

HCOLOR **HCOLOR=4**

Sets high-resolution graphics color to the color specified by HCOLOR. Color names and their associated values are:

0 black1	**4** black2
1 green (depends on TV)	**5** (depends on TV)
2 blue (depends on TV)	**6** (depends on TV)
3 white1	**7** white2

HGR

Available only in Applesoft BASIC. Sets high-resolution graphics mode (280 × 160) for the screen, leaving four lines for text at the bottom. The screen is cleared to black, and page 1 of memory is displayed. Neither HCOLOR nor text screen memory is affected when HGR is executed. The cursor is not moved into the text window.

HGR2

Sets full-screen high-resolution graphics mode (280 × 192). The screen is cleared to black, and page 2 of memory is displayed. Text screen memory is not affected.

HIMEM: 16384

Sets the address of the highest memory location available to an Applesoft BASIC program, including variables. It is used to protect an area of memory for data, high-resolution screens, or machine-language routines. HIMEM: is not reset by CLEAR, RUN, NEW, DEL, changing or adding a program line, or reset.

HLIN **HLIN 10, 20 AT 30**

Used to draw horizontal lines in low-resolution graphics mode using the color most recently specified by COLOR. The origin (x=0 and y=0) for the system is the top leftmost dot of the screen. In the example, the line is drawn from x=10 to x=20 at y=30. Another way to say this: the line is drawn from (10, 30) through (20, 30).

HOME

Moves the cursor to the upper left screen position within the text window and clears all text in the window.

COMMAND | **EXAMPLE**

HPLOT

HPLOT 10,20
HPLOT 30,40 TO 50,60
HPLOT TO 70,80

Plots dots and lines in high-resolution graphics mode using the most recently specified value of HCOLOR. The origin is the top leftmost screen dot (x=0, y=0). The first example plots a high-resolution line from the dot at (x=10, y=20). The second example plots a high-resolution line from the dot at (x=30, y=40) to the dot at (x=50, y=60). The third example plots a line from the last dot plotted to the dot at (x=70, y=80), using the color of the last dot plotted, not necessarily the most recent HCOLOR.

HTAB

HTAB 23

Moves the cursor either left or right to the specified column (1 through 40) on the screen. In the example, the cursor will be positioned in column 23.

IF <condition>THEN <process>

IF AGE <18 THEN A=0: B=1: C=2
IF ANS$="YES" THEN GOTO 100
IF N < MAX THEN 25
IF N < MAX THEN GOTO 25

If the expression following IF evaluates as true (i.e., non-zero), then the instructions following THEN in the same line will be executed. Otherwise, any instructions following THEN are ignored, and execution passes to the instruction in the next numbered line of the program. String expressions are evaluated by alphabetic ranking. Examples 2, 3, and 4 behave the same, despite the different wordings.

INPUT

INPUT A%
INPUT "TYPE AGE THEN A COMMA THEN NAME";B,C$

In the first example, INPUT prints a question mark and waits for the user to type a number, which will be assigned to the integer variable A%. In the second example, INPUT prints the optional string exactly as shown, then waits for the user to type a number (which will be assigned to the real variable B), then a comma, and then string input (which will be assigned to string variable C$). Multiple entries to INPUT may be separated by commas or returns.

INT

INT(NUM)

Returns the largest integer less than or equal to the given argument. In the example, if NUM is 2.389, then 2 will be returned; if NUM is −45.123345, then −46 will be returned.

INVERSE

Sets the video mode so that the computer's output prints as black letters on a white background. Use NORMAL to return to white letters on a black background.

COMMAND | **EXAMPLE**

IN# — **IN#4**

Specifies the slot (from 1 through 7) of the peripheral that will provide subsequent input for the computer. IN# 0 re-establishes input from the keyboard instead of the peripheral.

LEFT$(...) — **LEFT$("APPLESOFT II",5)**

Returns the specified number of leftmost characters from the string. In the example, APPLE (the 5 leftmost characters) will be returned.

LEN — **LEN("AN APPLE A DAY")**

Returns the number of characters in a string, between 0 and 255. In the example, 14 will be returned.

LET — **LET A=23.567**
A$="DELICIOUS"

The variable name to the left of = is assigned the value of the string or expression to the right of the =. The LET is optional.

LIST — **LIST**
LIST 200-3000
LIST 200,3000
LIST ,3000
LIST 3000,

The first example causes the whole program to be displayed on the TV screen. The second example causes program lines 200 through 3000 to be displayed. To list from the start of the program through line 200, use LIST -200. To list from line 200 to the end of the program, use LIST 200-. The third example behaves like the second example. LISTing is aborted by CTRL-C.

LOAD

Reads Applesoft BASIC program from disk.

LOG — **LOG(2)**

Returns the natural logarithm of the specified arithmetic expression. In the example, .693147181 is returned.

LOMEM: 2060

Sets the address of the lowest memory location available to a BASIC program. This allows protection of variables from high-resolution graphics in computers with large amounts of memory.

MID$(...) — **MID$("AN APPLE A DAY",4)**
MID$("AN APPLE A DAY",4,9)

Returns the specified substring. In the first example, the fourth through the last characters of the string will be returned: APPLE A DAY. In the second example, the nine characters beginning with the fourth character in the string will be returned: APPLE A D.

COMMAND	**EXAMPLE**

NEW
Deletes the current program and all variables.

NEXT
See the discussion of FOR...TO...STEP.

NORMAL
Sets the video mode to the usual white letters on a black background for both input and output.

NOTRACE
Turns off the TRACE mode. (See TRACE.)

ON exp GOSUB line1,line2, ..., etc. — **ON ID GOSUB 100,200,4005,500**
Executes a GOSUB to the line number indicated by the value of the arithmetic expression following ON. In the example, if ID is 1, GOSUB 100 is executed; if ID is 2, GOSUB 200 is executed, and so on.

If the value of the expression is 0, or is greater than the number of listed alternate line numbers, then program execution proceeds to the next statement.

ON ID GOTO — **ON ID GOTO 100,200,23,4005,500**
Identical to ON ID GOSUB (see above), but this command executes a GOTO branching to the line number indicated by the value of the arithmetic expres sion following ON.

ONERR GOTO — **ONERR GOTO 500**
Used to avoid an error message that halts execution when an error occurs. When executed, ONERR GOTO sets a flag that causes an unconditional jump to the indicated line number (500, in the example), if any error is later encountered.

PDL(3)
Returns the current value, a number from 0 through 255, of the indicated game control paddle. Game paddle numbers 0 through 3 are valid.

PEEK — **PEEK(37)**
Returns the contents, in decimal, of the byte at the specified decimal address (37 in the example).

PLOT — **PLOT 10,20**
In low-resolution graphics mode, places a dot at the specified location. In the example, the dot will be at (x=10, y=20). The color of the dot is determined by the most recent value of COLOR, which is 0 (black), if not previously specified.

POKE — **POKE −16302,0**
Stores the binary equivalent of the second argument (0, in the example) into the memory location whose decimal address is given by the first argument (−16302, in the example).

COMMAND	**EXAMPLE**

POP
Causes one RETURN address to "pop" off the top of the stack of RETURN addresses. The next RETURN encountered after a POP causes a branch to one statement beyond the second most recently executed GOSUB.

POS — **POS(0)**
Returns the current horizontal position of the cursor. This is a number from 0 (at the left margin) to 39 (at the right margin). What you put inside the parentheses is unimportant, if it can be evaluated by Applesoft BASIC.

PRINT — **PRINT**
PRINT A$; "X=";X
The first example causes a linefeed and return to be executed on the screen. Items in a list to be PRINTed should be separated by commas if each is to be displayed in a separate tab field. The items should be separated by semicolons if they are to be printed next to each other without any intervening space. If A$ contains "CORE", and X is 3, the second example causes COREX=3 to be printed.

PR# — **PR #2**
Transfers output to the specified slot, 1 through 7. PR#0 returns output to the TV screen.

READ — **READ A, B%, C$**
When executed, assigns the variables in the READ statement successive values from elements in the program's DATA statements. In the example, the first two elements in the DATA statements must be numbers, and the third a string (which may be a number). They will be assigned, respectively, to the variables A, B%, and C$.

REM — **REM THIS IS A REMARK**
Allows text to be inserted into a program as remarks.

RESUME
At the end of an error-handling routine (see ONERR GOTO), causes the resumption of the program at the statement in which the error occurred.

RETURN
Branches to the statement immediately following the most recently executed GOSUB.

RIGHT$(...) — **RIGHT$("SCRAPPLE",5)**
Returns the specified number of rightmost characters from the string. In the example, APPLE (the 5 rightmost characters) will be returned.

RND — **RND(1)**
Returns a random real number greater than or equal to 0 and less than 1. RND(0) returns the most recently generated random number. Each negative

COMMAND **EXAMPLE**

argument generates a particular random number that is the same every time RND is used with that argument, and subsequent RNDs with positive arguments will always follow a particular, repeatable sequence. Every time RND is used with any positive argument, a new random number from 0 to 1 is generated, unless it is part of a sequence of random numbers initiated by a negative argument.

ROT — **ROT=16**

Sets angular rotation for a shape to be drawn by DRAW or XDRAW. ROT=0 causes a shape to be DRAWN oriented just as it was defined. ROT=16 causes a shape to be DRAWn rotated 90 degrees clockwise, and so forth. The process repeats starting at ROT=64.

RUN — **RUN 500**

Clears all variables, pointers, and stacks and begins execution at the indicated line number (500 in the example). If no line number is specified, execution begins at the lowest-numbered line in the program.

SAVE

Stores a program on disk.

SCALE — **SCALE=50**

Sets the scale size for a shape to be drawn by DRAW or XDRAW, SCALE=1 sets the point for point reproduction of the shape definition. SCALE=255 results in each plotting vector being extended 255 times. NOTE: SCALE=0 is the maximum size and not a single point.

SCRN — **SCRN(10,20)**

In low-resolution graphics mode, returns the color code of the specified point. In the example, the color of the dot at (x=10, y=20) is returned.

SGN — **SGN(NUM)**

Returns −1 if the argument is negative, 0 if the argument is 0, and 1 if the argument is positive.

SIN — **SIN(2)**

Returns the sine of the argument, which must be in radians. In the example, .909297427 is returned.

SPC — **SPC(8)**

This instruction, when used, is to be used only within a PRINT statement. It is optional. It introduces the specified number of spaces (8, in the example) between the last item PRINTed and the next item PRINTed if semicolons precede and follow the SPC command.

SPEED — **SPEED=50**

Sets the rate at which characters are to be sent to the screen or other input/output devices. The slowest rate is 0, the fastest is 255.

COMMAND **EXAMPLE**

SQR **SQR(2)**
Returns the positive square root of the argument. In the example, 1.41421356 is returned. SQR executes more quickly than ^.5.

STR$(...) **STR$(12.45)**
Returns a string that represents the value of the argument. In the example, the string "12.45" is returned.

TAB **TAB(23)**
Must be used in a PRINT statement. The argument must be between 0 and 255 and enclosed in parentheses. For arguments 1 through 255, if the argument is greater than the value of the current cursor position, then TAB moves the cursor to the specified printing position, counting from the left edge of the current cursor line.

If the argument is less than the value of the current cursor position, then the cursor is not moved. TAB(0) puts the cursor into position 256.

TAN **TAN(2)**
Returns the tangent of the argument, which must be in radians. In the example, −2.18503987 is returned.

TEXT
Sets the screen to the usual nongraphics text mode with 40 characters per line and 24 lines. Also resets the text window to full screen.

TRACE
Causes the line number of each statement to be displayed on the screen as it is executed. TRACE is not turned off by RUN, CLEAR, NEW, DEL, or reset. NOTRACE turns off TRACE.

USR **USR(3)**
This function passes its argument to a machine-language subroutine. The argument is evaluated and put into the floating-point accumulator (locations $9D through $A3), and a JSR to location $0A is performed. Locations $0A through $0C must contain a JMP to the beginning location of the machine-language subroutine. The return value for the function is placed in the floating-point accumulator. To return to Applesoft BASIC, do an RTS.

VAL **VAL("−3.7E4A5PLE")**
Attempts to interpret a string up to the first non-numeric character as a real or an integer, and returns the value of that number. If no number occurs before the first non-numeric character, a zero is returned. In the example, −37000 is returned.

VLIN **VLIN 10,20 AT 30**
In low-resolution graphics mode, draws a vertical line in the color indicated by the most recent COLOR statement. The line is drawn in the column

COMMAND **EXAMPLE**

indicated by the third argument. In the example, the line is drawn from y=10 to y=20 at x=30.

VTAB **VTAB 15**

Moves the cursor to the line on the screen specified by the argument. The top line is line 1. The bottom line is line 24. VTAB moves the cursor up or down but not left or right.

WAIT **WAIT 16000,255**

Allows a conditional pause to be inserted into a program. The first argument is the decimal address of a memory location to be tested to see when certain bits are high (1, or on) and certain bits are low (0, or off).

Each bit in the binary equivalent of the second decimal argument indicates whether you're interested in the corresponding bit in the memory location: 1 means you're interested, 0 means ignore that bit.

Each bit in the binary equivalent of the third decimal argument indicates which state you're WAITing for the corresponding bit in the memory location to be in: 1 means the bit must be low, 0 means the bit must be high. If no third argument is present, 0 is assumed.

If any one of the bits indicated by a 1-bit in the second argument matches the state for that bit indicated by the corresponding bit in the third argument, the WAIT is over.

XDRAW **XDRAW 3 AT 180,120**

In the example, draws shape definition 3 from a previously loaded shape table in high-resolution graphics beginning at (x=180, y=120). For each point plotted, the color is the complement of the color already existing at that point. Provides an easy way to erase. If you XDRAW a shape then XDRAW it again, you'll erase the shape without erasing the background.

INDEX

Documentation to Accompany the Optional Program Disk—Apple //c Explorer's Disk

The *Apple //c Explorer's Disk* is available as an option to this book. It provides a number of useful tools and further Applesoft BASIC programming lessons and examples. It was developed specifically for the Apple //c and the printers most likely to be used with it: the Apple Imagewriter and Scribe printers. The disk offers an initial menu providing access to the three Utilities and five Applesoft tutorials in exactly the same way that functions are selected from the *System Utilities* disk menus. After making the first selection, subsequent menus and onscreen directions make using the programs very easy. Additional documentation may be displayed on the screen for each program. In addition, all of the programs may be listed for further study as programming examples. The Utilities may be moved to other disks where they may be more convenient for you to use them.

Imagewriter Set-up Program

This program makes all of the commonly used advanced features of the Apple Imagewriter printer available from a menu. When a selection is made as to the type or size of characters, line spacing, or other characteristics, the printer is given appropriate instructions to produce the desired print style. A sample may then be printed to verify that the printer is properly prepared. At this point material from other programs may be printed in this same style. Some programs will reinitialize the Imagewriter to its default mode, in which case all printer control must be provided by the program. Most programs do not reset the printer, and will thus continue printing in the style established from the menu. This entire program, or selected parts of it, may also be incorporated into your own programs to provide easy access to a variety of print styles. The features controlled include:

Line Spacing
Character Size
Emphasized Printing

Scribe Set-up Program

This program provides functions similar to the Imagewriter Set-up Program, but for Apple's revolutionary new Scribe Printer. It provides simple menu-driven access to the most commonly used advanced features of the Scribe, such as:

Draft Mode (80 CPS)
Near Letter Quality (NLQ) Mode (50 CPS)
Superscript
Subscript
Line Spacing
Left Margin Set
10 Characters Per Inch (80 Characters Per Line)
17 Characters Per Inch (136 Characters Per Line)

ProSNOOP Disk Utility

ProSNOOP provides you with the ability to see exactly what is stored on a ProDOS disk on a sector-by-sector, byte-by-byte basis. This is interesting if you want to see how ProDOS organizes the disk in detail. It can also be valuable if you should have a disk go bad without a backup (but do not let this happen!). You could use ProSNOOP to view the contents of any sectors remaining intact within damaged files. ProSNOOP does not allow you to change the disk or to prevent damage to your files. ProSNOOP uses a menu system exactly like the *System Utilities* and output may be directed to either the screen or the printer.

Applesoft BASIC Tutorials

These five sample Applesoft BASIC programs provide a series of gradually more advanced programming examples. Each is documented to show the purpose, how the purpose was translated into Applesoft BASIC statements, how the finished program works, and what the underlying principles are. The programs are designed to be run with data you supply so you can grasp their operation and usefulness. They are also designed to provide sgements of BASIC programming that you can "lift" into your own programs to simplify the programming process. Alternatively, you can adapt these programs as "shells," adding statements to transform these programs into the programs you desire.

The tutorial programs are carefully documented, both within the program listing with REMARK statements and with onscreen prompts designed to explain the functioning of the program as it runs. They offer an ideal opportunity to continue the process of "getting your feet wet" with BASIC and move you closer to achieving competence in programming on your own.

REFERENCE CARD

ABBREVIATIONS

addr	=	address in memory
aexp	=	arithmetic expression
char.	=	character
expr	=	expression
linenum	=	line number
ln#	=	line number
sexpr	=	string expression
var	=	variable
x,y	=	an x,y coordinate
[ ]	=	enclosed elements may be omitted
( )	=	enclosed elements may be repeated one or more times
|	=	separates alternative definitions

MATHEMATICAL OPERATORS

= assigns value to a variable
+ addition
− negation (subtraction)
* multiplication
/ division
exponentiation

LOGICAL OPERATORS

< greater than
> less than
>= less than or equal to
<= greater than or equal to
<> not equal

APPLESOFT COMMANDS

SYNTAX	*EXAMPLE*	*DESCRIPTION*
ABS(aexp)	**ABS(-3.14159)**	Yields the absolute value of the argument.
ASC(sexpr)	**ASC("QUITE")**	Yields the ASCII code for the first character in the argument.
ATN(AEXP)	**ATN(2)**	Yields the arc tangent, in radians.
CALL addr	**CALL-922**	Executes the machine language subroutine at the specified decimal memory address.
CHR$(aexp)	**CHR$(65)**	Yields the character corresponding to the ASCII code of the argument.
CLEAR	**CLEAR**	Resets all variables and internal control information to their initial state.
COLOR =aexp	**COLOR=15**	Sets the display color for plotting low-resolution graphics. See Appendix B.
CONT	**CONT**	Resume program execution after it has been halted by a STOP, END, or CTRL-C.
COS(aexp)	**COS(2)**	Yields the cosine of the argument, expressed in radians.
DATA [literal|	**DATA JOHN SMITH**	Creates a list of items for use by READ statements.
or **|string|**	**DATA "CODE 22"**	
or **|real|**	**DATA 15.75**	
or **|integer]**	**DATA -60**	
DEF FN name (name)= aexp	**DEF FN A(W) =2*W+W**	Defines a new function for use in the program.
DEL ln#,ln#	**DEL 2300,5600** **DEL ,100**	Deletes a range of consecutive numbers from the program.
DIM name	**DIM AGE(20,3)** **DIM N$(50), A%(5)**	Defines and allocates space for one or more arrays.
DRAW aexp	**DRAW 4** **DRAW 4 AT 50,1000**	Draws a shape at a specified point on the high-resolution graphics screen.
END	**END**	Terminates the execution of the program and returns control to the user.
EXP(aexp)	**EXP(2)**	Yields the mathematical exponential of its argument.
FLASH	**FLASH**	Causes all the text displayed on the screen by subsequent PRINT statements to flash.
FN name (aexp)	**FN A(X)**	Applies a designated function to the value of the argument expression.
FOR name = aexp TO (step aexp) *e.g.* **FOR W=1 TO 20** **FOR Q=2 TO-3 STEP -2** **FOR X=1 TO 4 STEP 3**		Marks the beginning of a loop, identifies the index variable, and gives the variables starting and ending values.
FRE(expr)	**FRE(0)**	Yields the amount of remaining memory, in bytes, available to the program.
GET var	**GET ANS$**	Accepts a single character from the keyboard without displaying it on the screen.
GOSUB linenum	**GOSUB 250**	Executes a subroutine beginning at the designated line number.
GOTO linenum	**GOTO250**	Send control unconditionally to the designated line number.
GR	**GR**	Converts the display from text to low-resolution graphics.
HCOLOR =aexp	**HCOLOR =4**	Sets the display color for plotting high-resolution graphics. See Appendix B.

DISK OPERATING SYSTEM COMMANDS

(DOS 3.3 and ProDOS)

HOUSEKEEPING COMMANDS	*EXAMPLES*
*INIT f [,Vv][,Ss][,Dd]	INIT HELLO, V18
CATALOG [,Ss][,Dd]	CATALOG
SAVE f [,Ss][][Dd][,Vv]	SAVE COLOR DEMOS,V56
LOAD f [,Ss][,Dd][,Vv]	LOAD DOW JONES, V19,D1
RUN f [,Ss][,Dd][,Vv]	RUN ANNUITY,D2
RENAME f,g [,Ss][,Dd][,Vv]	RENAME SMALL, LARGE S4,D1,V0
DELETE f [,Ss][,Dd][,Vv]	DELETE TEST
LOCK f [,Ss][,Dd][,Vv]	LOCK LOVE LETTERS,V31
UNLOCK f [,Ss][,Dd][,Vv]	UNLOCK RECIPES,V31,D2
VERIFY f [,Ss][,Dd][,Vv]	VERIFY SAM
*MON [C][,I][,O]	MON O MON C,I,O
*NOMON [C][,I][,O]	NOMON O NOMON C, I, O
*MAXFILES n	MAXFILES 6

ACCESS COMMANDS	*EXAMPLES*
*FP	FP
*INT	INT
PR# s	PR# 6
IN# s	IN# 6
CHAIN f [,Ss] [,Dd] [,Vv]	CHAIN PART TWO,D1,S7,V0

SEQUENTIAL TEXT FILE COMMANDS	*EXAMPLES*
OPEN f, [,Ss] [,Dd] [,Vv]	OPEN SESAME,D2
CLOSE [f]	CLOSE WINDOW
WRITE f [,Bb]	WRITE ADDRESS.DATA
READ f [,Bb]	READ SESAME
APPEND f [,Ss] [,Dd] [,Vv]	APPEND MORE INFO
POSITION f[,Rp]	POSITION ADDRESS.DATA,R277
EXEC f[,Rp] [,Ss] [,Dd] [,Vv]	EXEC UTILITY

---

*DOS 3.3 command only. All other commands are DOS 3.3 and ProDOS.

All ProDOS commands require complete pathnames unless a prefix has been set. All parameters not in [ ] are required.

ProDOS OPERATING SYSTEM COMMANDS

(These commands are not the same as DOS 3.3 commands.)

COMMANDS	EXAMPLES
CAT pn [,Ss][,Dd]	CAT/UTILITIES
CREATE pn [,Tt][,Ss][,Dd]	CREATE/UTILITIES/PROGRAMS/ANIMALS
FLUSH pn or FLUSH	FLUSH/UTILITIES/PROGRAMS/ANIMALS
PREFIX pn (partial)	PREFIX/UTILITIES/PROGRAMS
RESTORE pn [,Ss][,Dd]	RESTORE/UTILITIES/PROGRAMS/ANIMALS
STORE pn [,Ss][,Dd]	STORE/UTILITIES/PROGRAMS/ANIMALS
- (dash) - pn [,Ss][,Dd]	-/UTILITIES/PROGRAMS/ANIMALS

HGR	**HGR**	Converts the high-resolution graphics display to page #1.
HGR2	**HGR2**	Converts the high-resolution graphics display to page #2.
HIMEM: aexp	**HIMEM: 16384**	Sets the address of the highest memory location available to the Applesoft II BASIC program.
HLIN aexpl, aexp2 AT aexp3	**HLIN 10,20 AT 30**	Draws a horizontal line in low-resolution graphics.
HOME	**HOME**	Clears all text from the text window and moves the cursor to the top left corner.
HPLOT aexp,aexp	**HPLOT 10,20**	Plots a point on the high-resolution graphics screen.
HPLOT aexp,aexp TO aexp,aexp e.g. **HPLOT 30,40 TO 50,60**		Draws a line on the high-resolution graphics screen.
HPLOT TO aexp,aexp e.g. **HPLOT TO 70,80**		Draws a line from the last point plotted to the new position.
HTAB aexp	**HTAB 23**	Positions the cursor to the specified column of the argument.
IF expr THEN statement e.g. **IF AGE<18 THEN A=0**		Executes the statement if the expression is true.
IF expr THEN linenum e.g. **IF N<MAX THEN 25**		Executes the line number branch if the expression is true.
IF expr THEN (GOTO) linenum e.g.**IF N<MAX GOTO 25** e.g. **IF ANS$="YES" THEN GOTO 100**		Executes the GOTO statement if the expression is true.
IN#aexp	**IN#4**	Specifies the source of the subsequent input.
INPUT [sexpr;]var[(,var)] e.g. **INPUT "ENTER AGE = ";A%**		Reads a line of input from the current input device.
INPUT var	**INPUT B**	Reads a line of input.
INT(aexp)	**INT(NUM)**	Yields the integer part of the argument value.
INVERSE	**INVERSE**	Causes all text displayed on the screen with subsequent print statements to appear as black on white.
LEFT$(sexpr,aexp) e.g. **LEFT$("APPLESOFT II",5)**		Yields a specified number of characters from the beginning of a string.
LEN(sexpr)	**LEN("APPLE" ANNIE)**	Yields the length of a string in characters.
LET		**LET A=23.567** Assigns the value of the expression following the = to the variable preceding the =.
LIST **LIST ln#-ln#** **LIST ln#,ln#**	**LIST** **LIST 200-3000** **LIST 200,3000**	Displays all or part of the program on the screen or writes it to the current output device.
LIST ,ln#	**LIST ,3000**	Displays all lines through line #.
LIST ln#,	**LIST 200,**	
LIST ln#	**LIST 222**	Displays a line number.
LOAD [name]	**LOAD DUMMY**	Reads a program into memory from a diskette or tape.
LOG(aexp)	**LOG(2)**	Yields the natural logarithm of the argument.
LOMEM:aexp	**LOMEM: 32767**	Sets the address of the lowest memory location available to the program for variable storage.
MID$ (sexpr,aexp) e.g. **MID$ ("AN APPLE A DAY",4)**		Yields the characters beginning at the specified position in a string.
MID$(sexpr, aexp,aexp) e.g. **MID$ ("AN APPLE A DAY",4,9)**		Yields a specified number of characters beginning at the specified position in the string.
NEW	**NEW**	Clears the current program from memory.
NEXT [var]	**NEXT I**	Marks the end of the FOR loop.
NORMAL	**NORMAL**	Causes all text displayed on the screen with a subsequent PRINT statement to appear as white on black.
NOTRACE	**NOTRACE**	Stops the display of executed line numbers.
ON aexp GOSUB linenum [[,linenum]] e.g. **ON ID GOSUB 100,** 200, 400 e.g. **ON NR GOSUB 50,100, 1000**		Chooses a subroutine to execute depending on the value of an expression.
ON aexp GOTO linenum [[,linenum]] e.g. **ON ID GOTO 100, 1000, 20** e.g. **ON NR GOTO 50,100,1000**		Chooses the line number to branch to depending on the value of an expression.
ONERR GOTO LN#	**ONERR GOTO 500**	Replaces the normal error-handling mechanism with a programmer-written routine.
PDL(aexp)	**PDL(3)**	Reads the current dial setting on the designated hand-controller.
PEEK(addr)	**PEEK(36)**	Yields the contents of the specified memory location.
PLOTx,y	**PLOT 10,20**	Plots a single block at the specified position on the low-resolution screen.
POKE addr,aexp	e.g. **POKE -16302,0**	Stores a value into the specified memory location.
POP	**POP**	Removes the most recent return address from the control stack.
POS(expr)	**POS(O)**	Yields the current position of the cursor on the text display.

PR#aexp	**PR#2**	Specifies the destination for subsequent output.
PRINT	**PRINT A$,"X=";X**	Writes a line of output to the current output device.
READ var	**READ A,B%,C$**	Reads values from DATA statements.
REM (char.)	**REM REMARK HERE**	Includes remarks in the body of a program.
RESTORE	**RESTORE**	Causes the next READ statement executed to begin reading at the beginning of the first DATA item.
RESUME	**RESUME**	Causes resumption of the program at the end of an ONERR GOTO routine.
RETURN	**RETURN**	Returns control from a subroutine.
RIGHT$(sexpr,aexp) e.g. **RIGHT$("SCRAPPLE",5)**		Yields a specified number of characters from the end of string.
RND (aexp)	**RND(1)**	Yields a random number from zero to one.
ROT=aexp	**ROT=16**	Sets the angular rotation for high-resolution shapes.
RUN	**RUN**	Executes an Applesoft II program.
	RUN 500	Executes a program at line number.
SAVE	**SAVE**	Writes a program to cassette tape.
	SAVE DUMMY	Writes a program to a diskette.
SCALE=aexp	**SCALE=50**	Sets the scale factor for a high-resolution shape.
SCRN(x,y)	**SCRN(10,20)**	Yields the code for the color currently displayed at the designated position.
SGN(aexp)	**SGN(NUM)**	Yields a value of −1,0, or +1 depending on the sign of the argument.
SIN(aexp)	**SIN(2)**	Yields the sine of the argument, expressed in radians.
SPC(aecp)	**SPC(8)**	Introduces the specified number of spaces in the line being printed.
SPEED=aexp	**SPEED=50**	Sets the rate at which text is scrolled on the display.
SQR(aexp)	**SQR(4)**	Yields the positive square root of the argument.
STOP	**STOP**	Terminates execution of a program.
STORE	**STORE XD**	Stores values from an array to disk.
STR$(aexp)	**STR$(12.45)**	Yields a string representing the numeric value of the argument.
TAB(aexp)	**TAB(23)**	Positions the text cursor to a specified position on the output line.
TAN(aexp)	**TAN(2)**	Yields the tangent of the argument.
TEXT	**TEXT**	Converts the display to 24 lines of text.
TRACE	**TRACE**	Causes the line number of each statement executed to be displayed as it is executed.
USR(aexp)	**USR(3)**	Executes a machine-language subroutine supplied by the user.
VAL(sexpr)	**VAL("-2.7E4")**	Yields the numeric value represented by the string.
VLIN aexp,aexp AT aexp e.g. **VLIN 10,20 AT 20**		Draws a vertical line in low-resolution graphics.
VTAB aexp	**VTAB 15**	Positions the cursor to a specified row of the text display.
WAIT aexp,axep	**WAIT 1600,255** **WAIT 1600,255,0**	Suspends program execution until a specified bit pattern appears at the memory location specified.
XDRAW aexp AT x,y e.g. **XDRAW 3 AT 180,120**		Draws a shape from the shape table in memory using the complement color.

ERROR MESSAGES

APPLESOFT ERROR MESSAGES

CODE	ERROR MESSAGE
0	NEXT WITHOUT FOR
16	SYNTAX ERROR
22	RETURN WITHOUT GOSUB
42	OUT OF DATA
53	ILLEGAL QUANTITY
69	OVERFLOW
77	OUT OF MEMORY
90	UNDEFINED STATEMENT
107	BAD SUBSCRIPT
120	REDIMENSIONED ARRAY
133	DIVISION BY ZERO
163	TYPE MISMATCH
176	STRING TOO LONG
191	FORMULA TOO COMPLEX
224	UNDEFINED FUNCTION
254	BAD RESPONSE TO INPUT STATEMENT
255	CTRL-C INTERRUPT ATTEMPTED
	CAN'T CONTINUE
	ILLEGAL DIRECT COMMAND

RANDOM ACCESS TEXT FILE COMMANDS	*EXAMPLES*
OPEN f, Lj [,Ss] [,Dd] [,Vv]	**OPEN SESAME,L2**
CLOSE [f]	**CLOSE** **CLOSE BOOK**
WRITE f [,Rr] [,Bb]	**WRITE ADDRESS.DATA,R3**
READ f [,Rr] [,Bb]	**READ SESAME,R3,B30**

MACHINE LANGUAGE FILE COMMANDS	*EXAMPLES*
BSAVE f, Aa, Lj [,Ss] [,Dd] [,Vv]	**BSAVE PICTURE, A16384,L8192** **BSAVE PICTURE, A$4000,L$2000**
BLOAD f [,Aa] [,Ss] [,Dd] [,Vv]	**BLOAD PICTURE,A8192** **BLOAD PICTURE, A$2000,S6,D1,V254**
BRUN f [,Aa] [,Ss] [,Dd] [,Vv]	**BRUN SUPER, A$SCOA,S6,D2,V75**

MINI-ASSEMBLER COMMANDS

CALL -151	Places you into Monitor from BASIC
F666G	Turns on mini-assembler
FF69G	Turns off mini-assembler
addr	Examines value at that address
addr.addr	Examines values between addresses
addr:number	Stores hexadecimal number at memory address
addr3 < addr1.addr2M	Moves memory contents from address 1 thru address 2 to address 3
addr3 < addr1.addr2V	Compares ranges of memory locations
addrG	Executes machine code at address
addrL	Disassembles machine code starting at address
addrS	Single steps through a machine code program
addrT	Traces through each machine code program.
I	Inverse video
N	Normal video
CTRL-E	Displays 6502 registers
CTRL-Y	Jumps to subroutinue at $3F8
SLOT# CTRL-P	Outputs to slot #
SLOT# CTRL-K	Accepts input from slot #

Beginners! Here's Everything You Need To Know About The Remarkable New Apple. . .From Opening The Box To Learning ProDOS And Applesoft BASIC!

APPLE *II*c USER GUIDE

Gary Phillips and Donald Scellato

Now—a guide that offers you the most comprehensive, jargon-free approach to learning the Apple *II*c on the market today! Designed for beginners—and particularly ideal for family use—APPLE *II*c USER GUIDE presents a clear account of the programming capabilities and applications for the Apple *II*c. You'll love the friendly, step-by-step instructions delivered throughout—from the initial guided tour on how to set up your Apple *II*c system to the useful appendices that clue you in to the names and addresses of Apple user groups, vendors, and trade publications!

Designed for those with little or no previous computer experience, this "hands-on " tutorial includes practical discussions on:

★ Applesoft BASIC
★ ProDOS and the System Utilities
★ Selecting software and hardware
★ Monitors, printers, disk drives, modems, and much more!

CONTENTS:

ALSO AVAILABLE. . .OPTIONAL DISKETTE

This diskette contains the tools and examples used throughout the book. Also includes:

★ Additional Applesoft BASIC programming examples
★ Programs for viewing the contents of ProDOS disks on a sector-by-sector basis
★ Plus—programs for setting up the Imagewriter and Scribe printers for special printing!

ISBN 0-89303-307-3